OPTIMIZATION STUDIES OF TWO WATER PURIFICATION SYSTEMS

763

by

THOMAS DEAN DENCHFIELD

B. S., KANSAS STATE UNIVERSITY, 1965

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Chemical Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1967

Approved by:

Liang-tseng Fan

Major Professor

ACKNOWLEDGEMENTS

The author is indebted to many people who assisted with this work.

The Chemical Engineering Faculty, under the direction of Professor Wm. H. Honstead, Head, provided graduate course work which was most useful in preparation for writing this work.

Dr. L. V. Sorg and Mr. J. C. Lamkin of the American Oil Company supplied written material on the Sugar Creek aerated lagoon. Also, discussions with these gentlemen about aerated lagoon processes were invaluable.

The Kansas State University Computing Center Staff is thanked for running many of the computer programs used in this work. Dr. E. S. Lee is thanked for having the POP computer programs run.

The financial assistance provided by a National Science Foundation Traineeship made the author's graduate study possible, and it is greatly appreciated.

The help of Dr. L. E. Erickson who offered many suggestions which appear in the final copy of this work is gratefully acknowledged.

Finally, the enthusiasm, guidance, and encouragement imparted to the author by Professor L. T. Fan, his major advisor, during the author's graduate study and the writing of this work are sincerely appreciated.

TABLE OF CONTENTS

PART I. THE USE OF THE NONLINEAR PROGRAMMING TECHNIQUE, POP-II TO OPTIMIZE A MULTI-EFFECT MULTI-STAGE (MEMS) SEAWATER DISTILLATION PLANT

PART II. OPTIMIZATION OF A MULTI-STAGE AERATED LAGOON BY THE DISCRETE MAXIMUM PRINCIPLE

PART I.

THE USE OF THE NONLINEAR PROGRAMMING TECHNIQUE POP-II TO OPTIMIZE A MULTI-EFFECT MULTI-STAGE (MEMS) SEAWATER DISTILLATION PLANT

1.0 INTRODUCTION

In this part of the report a Multi-Effect Multi-Stage (MEMS) seawater distillation plant was optimized by applying the nonlinear programming technique known as POP-II. Several concepts basic to the understanding of nonlinear programming will be presented first.

1.1 Some Basic Concepts and Definitions Useful in Nonlinear Programming

Unfortunately no single algorithm exists, with the exception of a systematic exhaustive search, for solving the general nonlinear programming problem which has an objective function of the form

$$S = f(x_1, x_2 \ldots\ldots, x_s) = f(x) \qquad (1a)$$

$$x = (x_1, x_2, \ldots\ldots, x_s)$$

and constraints of the form

$$g_i(x) \begin{Bmatrix} \leq \\ = \\ \geq \end{Bmatrix} b_i \qquad (1b)$$

$$i = 1, 2, \ldots\ldots, m$$

It should be apparent that there is no single method which is best for attacking the above problem. Many times the best method for attacking a problem will depend a great deal upon the exact functional forms of the objective function and the constraints. For example, if all the constraints are equality constraints, the methods given by Hadley (1) can be applied.

A few special cases of the general nonlinear programming problem will be given in the sections that follow. Fortunately, most problems of practical interest can be handled by one or more of the special cases or by more general techniques such as POP-II.

Before proceeding further, certain definitions which are important to nonlinear programming will be introduced.

A bounded region of n dimensional space, sometimes called n-space or hyperspace, is a portion of hyperspace which has been closed off or surrounded. Hypersurfaces are used to form the bounded region. The concept of a bounded region is best visualized in terms of 2 or 3 dimensional space. Figure 1a shows a portion of the first quadrant which is bounded by plane curves. Figure 1b shows a portion of the first octant which is bounded by a spherical surface.

A point which is inside a bounded region is called an interior point. A point on the boundary is a surface point. A point outside of the bounded region is called an exterior point. Interior and surface points are feasible points whereas an exterior point is said to be infeasible.

The quadratic form is useful in nonlinear programming in that by examining the quadratic form of a function it is possible to make important statements about the nature of the function. A function is said to be strictly concave if its quadratic form is negative definite over all of Euclidean space. For a function with one variable this corresponds to the case in which there is a unimodal maximum. On the other hand, if the quadratic form of the function is positive definite, the function is said to be strictly convex. This corresponds to a function with a unimodal minimum in the one variable case. Convex and concave functions of a single variable are shown in Figures 2a and 2b, respectively.

Recall that if the quadratic form of a function, $h(x)$, is positive definite, then the quadratic form of $-h(x)$ is negative definite. This implies that if $h(x)$ is convex, then $-h(x)$ is concave and conversely. Or,

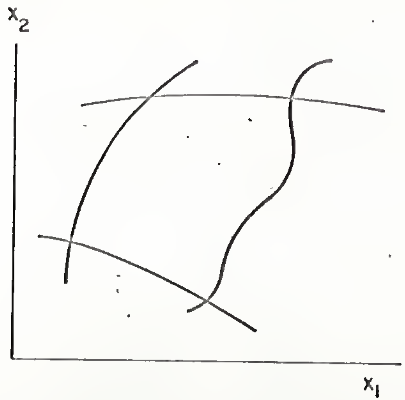

Fig. 1a. A portion of the first quadrant bounded by plane curves.

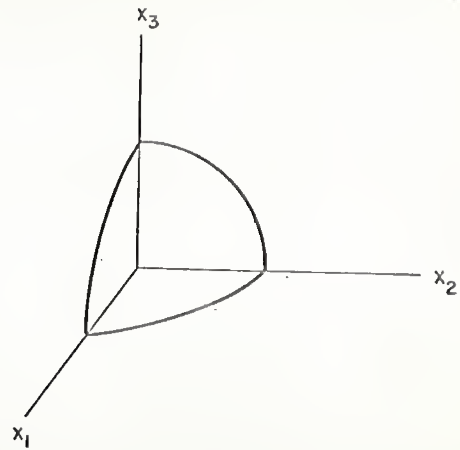

Fig. 1b. A portion of the first octant bounded by a spherical surface.

if h(x) is unimodal with a maximum, then -h(x) is unimodal with a minimum. That is, it is possible to change a minimization problem into a maximization problem or a maximization problem into a minimization problem. Therefore, one may talk exclusively about either maximization or minimization processes.

The following relations also hold, where

$$\text{(Convex Function)} \quad f_1\left[\lambda x_1+(1-\lambda)x_2\right] \leq \lambda f_1(x_1)+(1-\lambda)\, f_1(x_2) \qquad \text{(2a)}$$

$$\text{(Concave Function)} \quad f_2\left[\lambda x_1+(1-\lambda)x_2\right] \geq \lambda f_2(x_1)+(1-\lambda)\, f_2(x_2) \qquad \text{(2b)}$$

One way of thinking about the significance of relation, Equation (2a), is to consider the problem of approximating the convex function pictured in Figure 2a by a straight line through any two points on the curve. Observe that the straight line lies above the function it is approximating. That is, it over estimates the function. Similarly, the straight line approximation to a concave function always under estimates the function it is approximating. Relations given by Equations (2a) and (2b) can be generalized to higher spaces by replacing x by the s-dimensional vector, x = $(x_1, x_2, \ldots., x_s)$. In the general case of s-space these relations are often used to define convex and concave functions.

The important use of the definitions of convexity and concavity is to determine whether or not an extremum is a local or a global extremum. For the case of minimization of an objective function, a local minimum which occurs within a closed convex constraint set is a global minimum if the objective function is convex. A local maximum is a global maximum if the objective function is concave and the constraints are closed and convex. Note in both cases the constraint set must be closed and convex.

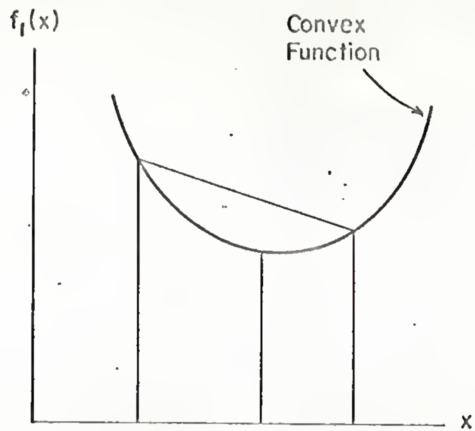

Fig. 2a. Convex function of a single variable.

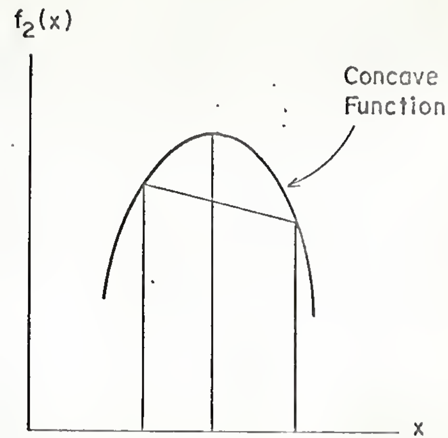

Fig. 2b. Concave function of a single variable.

For many problems, especially those with large numbers of constraint equations, it is quite a laborious task to determine if the objective function is convex or concave and if the constraint equations are convex. Therefore, what is generally done is to solve the problem without examining the object function and constraints with respect to convexity and concavity. Simulation and/or a search technique are then used to verify the results. If this reveals a solution point which gives a better extremum than the previous solution, the new point is used as a starting point for applying the particular nonlinear programming method again. The danger of a local extremum should be kept well in mind, but for many problems a local optimum is better than none at all.

The careful reader will also note that it is possible for an objective function to be convex over parts of the constraint region and concave over other parts. This case is shown in Figure 3.

Furthermore, some functions, such as, the one given by equation (3), may be neither convex nor concave.

$$S = \sum_{i=1}^{s} a_i x_i \qquad (3)$$

1.2 The Kuhn-Tucker Conditions

The Kuhn-Tucker conditions are useful because they give some insight into nonlinear programming theory and they enable one to determine whether or not a point which has been found by a computational procedure is an optimum point.

Kuhn and Tucker (2) were the first to derive the conditions which bear their names. Carr and Howe (3) also give a very readable derivation of the conditions.

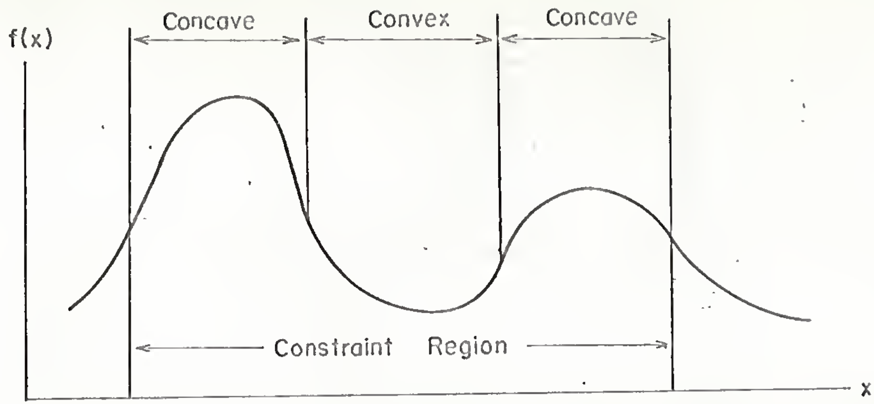

Fig. 3. Objective function which is both concave and convex in the constraint region.

The derivation of the Kuhn-Tucker conditions given here is due to Wilde and Beightler (4).

Consider the problem:

$$\min: \quad S = f(x) \tag{4a}$$

subject to the constraints

$$g_i\,(x) \geq b_i \tag{4b}$$

There is no loss in generality by using constraint equation (4b), in place of constraint equation (1b) since constraint equation (4b) can be obtained from constraint equation (1b) by introducing more slack variables and constraints.

Constraint equation (4b) can be charged to an equality constraint by introducing the slack variable, u, which is squared to insure that the squared quantity remains positive.

$$g_i\,(x) - b_i - u_i^2 = 0 \tag{5}$$

A Lagrangian function can be formed

$$L = f(x) - \sum_{i=1}^{m} \lambda_i \left[g_i\,(x) - b_i - u_i^2 \right] \tag{6}$$

Then, three of the four Kuhn-Tucker conditions which are necessary conditions for a minimum are:

$$\frac{\partial L}{\partial x_j} = 0 = \frac{\partial f(x)}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial g_i(x)}{\partial x_j}, \; j = 1, 2, \ldots, s \tag{7}$$

$$\frac{\partial L}{\partial \lambda_i} = 0 = -\left(g_i(x) - b_i - u_i^2 \right), \; i = 1, 2, \ldots, m \tag{8}$$

$$\frac{\partial L}{\partial u_i} = 0 = 2\lambda_i u_i, \; i = 1, 2, \ldots, m \tag{9}$$

The fourth necessary Kuhn-Tucker condition for a minimization problem as given by Wilde and Beightler (4) requires that the Lagrange Multipliers be

non-negative.

$$\lambda_i \geq 0, \quad i = 1, 2, \ldots, m \tag{10}$$

If a point does satisfy the Kuhn-Tucker conditions, it may or may not be a minimum. However, if it does not satisfy them, it can not possibly be a minimum. If both the objective function and the constraints are convex, the Kuhn-Tucker conditions are sufficient conditions for a global minimum. The Kuhn-Tucker conditions in themselves do not provide a computational procedure for finding optimal solutions to nonlinear programming problems.

2.0 NONLINEAR PROGRAMMING METHODS

2.1 The Classical Calculus

For some nonlinear programming problems, classical calculus techniques can be used to find the optimum extreme point. The steps for finding the optimum extreme point given the objective function, equation (1a), and the constraints, equation (1b), are as follows: Ignoring the inequality constraints, find the stationary points of the objective function. Determine which if any of the stationary points are within the feasible region. For those points in the feasible region determine at which point the objective function takes on an extreme value. Search the boundaries of the feasible region to determine if the interior extreme point is a global extremum.

An example illustrating this procedure is:

Find the maximum value of the objective function

$$S = 10x_1 + 20x_2 + x_1x_2 - 2x_1^2 - 2x_2^2 \tag{11}$$

which is subject to the constraints

$$0 \leq x_1 \leq 7 \tag{12}$$

$$0 \leq x_2 \leq 8 \tag{13}$$

The stationary points and the values of the objective function at these points are given in Table 1. For this problem the maximum is at the interior point. Note however, that if the goal would have been to minimize the objective function, the optimal point would have been on the boundary.

Several mathematical programming algorithms are extentions of the simplex method of linear programming. The reluctance to abandon the simplex method may be partially explained by noting that large sums of money have been invested in the development of computer codes to solve large scale linear programming problems since 1947 when the simplex method started its rapid development. Furthermore, experienced people are available who can cast a nonlinear programming problem into a linear programming format.

Two classes of nonlinear programming problems which can be solved by modified simplex algorithms are separable and quadratic programming problems.

2.2 Separable Programming Problems

The separable programming technique can be applied when the constraints are linear and it is possible to separate the objective function into a sum of functions each of which is a function of only one independent variable. The general separable programming problem has the following form.

Objective function:

$$f(x) = \sum_{i=1}^{s} f_i \, (x_i) \tag{14}$$

Constraint equations:

$$\sum_{j=1}^{s} a_{ij} \, x_j \begin{Bmatrix} \leq \\ = \\ \geq \end{Bmatrix} b_i ; \; i = 1, \ldots, m \tag{15}$$

2.3 Quadratic Programming Problems

When the objective function has the quadratic form

Table 1. Stationary Points and Values of the Objective Function for the Classical Calculus Example

Location of Stationary Point	x_1	x_2	S
Interior	4.0	6.0	80.0
Boundary	0.0	5.0	50.0
Boundary	7.0	6.8	63.1
Boundary	4.5	8.0	72.5
Boundary	2.5	0.0	12.5

$$f(x) = \sum_{i=1}^{s} a_i x_i + \sum_{i=1}^{s} \sum_{j=1}^{s} C_{ij} x_i x_j \tag{16}$$

and the constraints are linear,

$$\sum_{j=1}^{s} a_{ij} x_j \begin{Bmatrix} \leq \\ = \\ \geq \end{Bmatrix} b_i;\ i = 1, \ldots\ldots, m \tag{17}$$

quadratic programming methods can be used to extremize the objective function.

Separable programming, quadratic programming, and methods for solving other types of nonlinear programming problems are discussed in detail by several authors (1, 3, 4). In this work, little emphasis will be placed on methods for solving problems which have specific objective and constraint equation forms because more general algorithms exist for solving more general types of mathematical programming problems.

2.4 The Process Optimization Program (POP-II)

For many of the nonlinear programming algorithms, no "general", easy to use, computer program (computer code) exists for implementing the calculations for large scale problems. Because this is a serious disadvantage to using these methods, there has been a great impetus to develop a general computer program to handle a wide variety of nonlinear optimization problems, which would require a minimum of effort, knowledge, and time on the users part. One computer code for doing this which has been quite successful is the Process Optimization Program II (POP-II) which has been developed by Smith (5), of IBM's System Research Institute.

POP-II uses a truncated Taylor series to obtain a sectionally linearized linear programming problem from the nonlinear programming problem. To start the computations, the processes' performance and objective equations

are linearized. A linear programming problem is formed from these linear equations. The solution of the linear programming problem, hopefully, gives values of the independent variables that are closer to the optimum values. The above procedure is repeated until the optimum values of the independent variables have been determined.

POP-II is a nonlinear programming technique which is easy to use. All that is necessary for using the technique is a mathematical model of the process or system which the user wishes to optimize and a mathematical statement of the optimization objective (objective function). The model must be constructed so that the processes' dependent variables can be calculated once the independent variables have been specified. If the user can provide a model which will calculate the dependent variables given the independent variables, then POP may well be a good technique to use.

Some of the terms discussed in this section may be easier to understand if one refers to Appendix II where the computer output for a POP problem is given.

The process model or simulation program which the user provides is named SUBROUTINE MODEL. SUBROUTINE MODEL must be programmed in such a way that each of the processes' dependent variables, the y_i values, is written as a function of one or more of the processes' independent variables, the x_j values, and/or a function of one or more of the previously defined dependent variables. In effect each dependent variable must be written as a function of one or more of the independent variables.

The sectionally linearized linear programming technique used by POP is described in the following paragraphs.

Consider the objective function and the equality constraints of the

generalized nonlinear programming problem given by equations (1a) and (1b).

$$S = f(x_1, x_2, \ldots., x_s) \tag{18}$$

$$g_p\ (x_1, x_2, \ldots., x_s) = b_p,\ p = 1, 2, \ldots, n \tag{19}$$

If the inequality constraints are ignored for the present, the number of independent variables for the nonlinear programming problem posed by equations (18) and (19) is equal to (s-n). Let the number of independent variables be r, that is, r is equal to (s-n).

Let the i th dependent variable be given by Y_i. However, y_1, is reserved for the objective function S. It is now possible to rewrite equations (18) and (19) in the form

$$y_1 = f(y_2, y_3, \ldots., y_{n+1}, x_1, x_2, \ldots., x_r) \tag{20}$$

and

$$g_p\ (y_2, y_3, \ldots., y_{n+1}, x_1, x_2, \ldots., x_r) = b_p,$$

where

$$p = 1, 2, \ldots., n \tag{21}$$

Equations (20) and (21) represent respectively, the general functional forms of the objective function and the equality constraints. In most instances, an individual g_p will not be a function of all of the y_i, $i = 2, 3, \ldots., n+1$ and x_y, $y = 1, 2, \ldots., r$. For purposes of programming a SUBROUTINE MODEL, each equality constraint is solved for a particular y_i value. The equality constraints, having been solved for a particular y_i value, are next arranged in an orderly fashion, such that each y_i is a function of the other dependent variables $y_2, y_3, \ldots., y_{i-1}$ and the independent variables, that is,

$$y_2 = h_2\ (x)$$

$$y_3 = h_3\ (h_2(x), x) = h_3(x)$$

$$\vdots$$

$$y_i = h_i\,(h_2(x),\, h_3(x),\, \ldots\ldots,\, h_{i-1}\,(x),\, x) = h_i\,(x) \tag{22}$$

$$\vdots$$

$$y_{n+1} = h_{n+1}\,(h_2\,(x),\, h_3(x),\, \ldots\ldots,\, h_n(x),\, x) = h_{n+1}(x)$$

where the abbreviated notation x ' $(x_1, x_2m \ldots\ldots x_r)$ is used. After the values of $y_2, y_3, \ldots\ldots, y_{n+1}$ have been determined, y_1 is calculated using equation (20).

The dependent variables given by equation (22) may be approximated in a small region near a point x_o, denoted by $x_o = (x_{1_o}, x_{2_o}, \ldots\ldots, x_{r_o})$, by a truncated Taylor Series if a coordinate system is defined such that the distance from x_o in the i th direction is denoted by δx_i. Thus we can write

$$y_i\,(x_o + \delta x) \simeq h_i(x)\Big|_{x_o} + \sum_{j=i}^{j=r} \delta x_j \, \frac{\partial h_i}{\partial x_j}\Big|_{x_o} \tag{23}$$

where

$i = 1, 2, \ldots\ldots, n$

Equation (23) is linear in δx_j and can be simplified to give

$$y_i\,(x_o + \quad x) \simeq e_i + \sum_{j=1}^{j=r} a_{ij} \quad \delta x_j\,, \tag{24}$$

where

$i = 2, 3, \ldots\ldots, n+1$

$$e_i = h_i\,(x)\Big|_{x_o} \tag{25}$$

and

$$a_{ij} = \frac{\partial h_i}{\partial x_j}\Bigg|_{x_o} \tag{26}$$

A similar approximation holds for y_1 in the small region near x_o.

The set of linear equations obtained by linearizing the nonlinear equations may be used to form a linear programming (LP) problem. The LP problem is solved in a small region near x_o. In most instances the solution will give a new x_o point which gives an improved value of the objective function.

When inequality constraints are imposed they are introduced into the LP problem by the POP program after the user has inserted the maximum and/or minimum limits of a variable on the input data cards. The POP program handles the inequalities by automatically introducing the necessary slack variables which are required.

The size of the small region near x_o in which the LP problem is solved may be controlled by user-specified constants which are called move limits. The j th move limit is the maximum step size that is permitted for δx_j.

Since the y_i values obtained in the linear programming solution may deviate from the true y_i values, the new x_o values are used in SUBROUTINE MODEL to calculate the true y_i values. The user must also supply error limits for the dependent variables. These error limits are used by POP to minimize the deviations that occur from the true y_i values.

The procedure described up until now is called a loop and is repeated until an extreme value of the objective function is found. In the event that an extreme value is not found, the procedure is terminated when the maximum number of optimization loops which the user specifies has been exceeded.

Many nonlinear optimization techniques require the calculation of derivatives of the same type. A significant feature of POP is that the values of the first partial derivatives required, the a_{ij} values, are

calculated automatically by a subroutine of POP. The central difference technique which is used to calculate the partial derivatives is given as

$$a_{ij} = \left.\frac{\partial h_i}{\partial x_j}\right|_{x_o} \simeq \frac{h_i\Big|_{x_{j_o} + \Delta x_j} - h_i\Big|_{x_{j_o} - \Delta x_j}}{2\,\Delta x_j} \tag{27}$$

where the Δx_j values are specified by the user. This technique is illustrated graphically in Figure 4 for the case where h_i is a function of one independent variable, x_1. Note that h_i is shown as a smooth continuous curve in Figure 4. Obviously, if the function whose first partial derivative is to be computed by a numerical method has one or more discontinuities for example, then the calculated derivative may be a very poor approximation. This is a disadvantage inherent in the use of POP. However, on the other hand, techniques which require analytical expressions for first partial derivatives require logical statements in their computer programs for handling piecewise smooth functions (see Appendix I of Part II of this work for an example.)

As an optional feature, the subroutine which calculates the partial derivatives can also calculate new move limit values as the optimization progresses. The process is called making an adaptive move limit calculation. This feature or operating mode of POP may be used with some SUBROUTINE MODEL'S to reach the optimum in a fewer number of loops. However, with some SUBROUTINE MODEL'S the adaptive move limit calculation reduces the values of the move limits so much that POP shuts off before reaching the optimum. Smith (5) gives a detailed description of the adaptive move limit operating mode.

In writing the SUBROUTINE MODEL used to simulate the MEMS plant,

$$\left.\frac{\partial h_i}{\partial x_1}\right|_{x_{1_0}} \simeq \text{Slope of } \overline{AB}$$

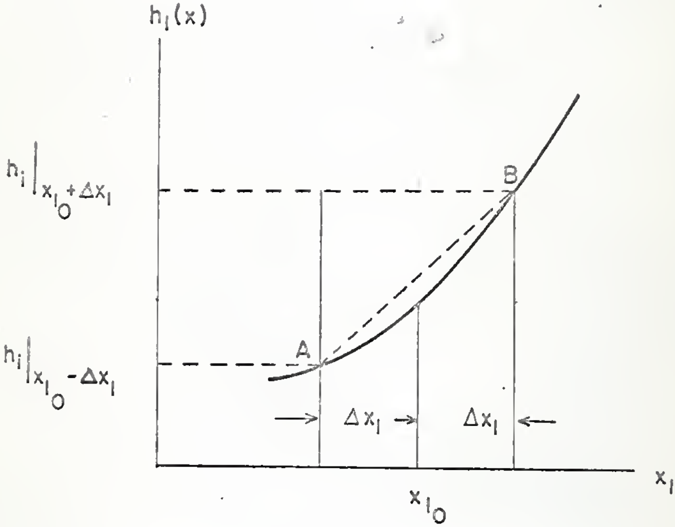

Fig. 4. Central difference approximation to the slope of the function $h_i\ (x_1)$ at the point x_{1_0}.

several procedures were found to help speed the programming and debugging processes. Since it is thought that these procedures will be helpful in general programming, they will be mentioned here.

In SUBROUTINE MODEL the j th independent variable is denoted as X(J). The i th dependent variable is denoted as Y(I). The i th constant is denoted as CONST(I). If large numbers of the symbols X(J), Y(I), and CONST(I) are used in a SUBROUTINE MODEL, it may be difficult to remember what each symbol represents. Consequently, it may be easier to debug SUBROUTINE MODEL by writing it in terms of easily recognizable symbols. The independent variables which are written in terms of X(J) variables at the beginning of the program. At the end of SUBROUTINE MODEL the Y(I) variable can be defined in terms of the easily recognizable symbols which are used in order to make the programming easier.

From experience with the MEMS plant problem, it was also determined that it is not necessary to define all dependent variables as Y(I) variables. For example, the horsepower requirements in the MEMS plant problem were not defined by Y(I) variables. Since the program, without modification, has a maximum limit of 50 Y(I) variables, this procedure may also help spread out the Y(I) variables and thus increase the problem size that can be handled.

The programming procedure described above will require more computer cards. However, from knowledge of the problem studied, it is felt that the additional computer cards in SUBROUTINE MODEL will not have a significant effect on the time required for execution of the POP program. Furthermore, the time from problem inception to completion may be greatly decreased.

One difficulty inherent in the use of POP, which also occurs in the

use of many other techniques, is the possibility of finding a relative or local extremum rather than a global extremum. If there are several local extreme points in the feasible answer space, POP will probably find the one which is nearest the starting point. This difficulty has been called "nearsightedness" by Baumol (6, 3). The way that "nearsightedness" is usually overcome is to start the program at several widely spaced feasible points and then observe what happens. Another problem which may be encountered when SUBROUTINE MODEL is large and/or has complex performance equations is that it may be difficult to find a feasible starting point. However, to aid the user, POP will double the move limits on successive loops until a feasible point has been found if one can be found at all. A diagram giving the steps required to solve a nonlinear programming problem using POP is given in Figure 4a.

One of the important features of POP is the matrix which is printed out after POP has reached an optimum point or optionally after each loop. Each element of this matrix, which is called the DYDX MATRIX, is one of the a_{ij} values given by equation (27). Specifically, the element a_{ij} is the number used in the i th row of the LP tableau with the variable x_j. For the i th row the y variable is y_i.

With the DYDX MATRIX, it is a simple matter to perform an incremental sensitivity analysis to determine where to concentrate effort to improve the process being optimized. In addition, inspection of the first row of the matrix, that is, the row for the objective variable y_1, will give a good indication of the nature of the response surface near the x_o point for which the matrix holds. This is especially true if the Δx_j values used are small.

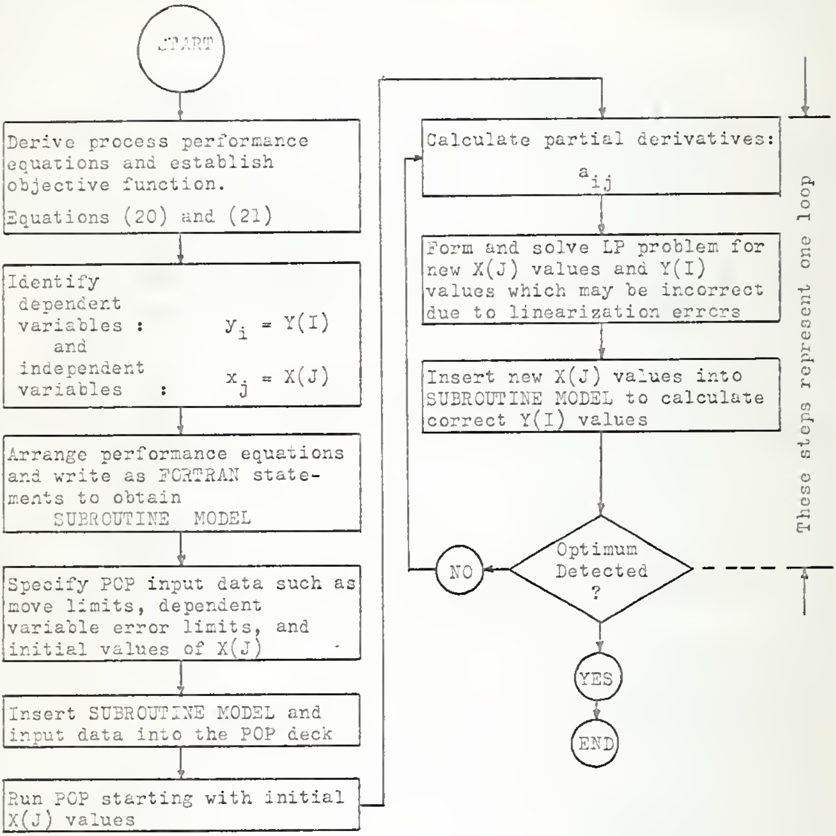

Fig. 4a. The steps required to solve a nonlinear programming problem using POP-II.

Each coefficient of the DYDX MATRIX can be examined with respect to sign and size. Consider the case when the coefficient for the i th row and the j th column is -a. This indicates that if the independent variable of the j th column is increased by one unit then the dependent variable of the i th row will be decreased by approximately a of its units. This analysis holds true only for a small region around the x_o point for which the matrix holds. Furthermore, the above statements are not strictly true for variables that appear in equality constraints.

If each element, a_{1j}, of the objective variable row of the matrix is a very small number, then small charges in the independent variables will not change the objective function significantly. If this situation occurs at the optimum, this is called a flat optimum and the response surface is said to be well behaved near the optimum.

However, if one of the elements, a_{1j}, of the objective variable row is very large, then only a small change in the independent variable of that element, x_j, will cause a large change in the objective function. This is the most interesting type of problem as it may be a constraint which causes this situation to happen. In the event that it is a constraint on a variable, which causes the element of the objective variable row to be large, one should look into the possibilities for removing or relaxing the constraint to some extent.

As an example, consider a problem in which reactor temperature appears as an independent variable. Furthermore, assume an upper limit is placed on the reactor temperature because the reactor cannot withstand high temperatures. Now if the system is optimized by POP and it is found that the reactor temperature element of the y_1 row of the DYDX MATRIX is the only large element in the row, and furthermore if this is due to the upper temperature imposed, then obviously searching for construction materials

which can withstand higher temperatures should be seriously considered. Admittedly this attack will necessitate resolving the problem with new cost data.

3.0 OPTIMIZATION OF A MEMS PLANT USING POP-II

An example problem of a Multi-Effect Multi-Stage (MEMS) seawater distillation plant which demonstrates the use of POP-II in process design will be given next.

3.1 Process Description

Fan, et. al. (7) give an excellent description of this process which is included here for convenience.

Figure 5 illustrates a three-effect multistage flash system. In order to facilitate the discussion, some critical locations in the system are denoted by letters, Z, A, B, C, D, etc. and the system is divided into various sections which are denoted by HR-1, R-1, HR-2, etc. The first effect consists of a brine heater H-1, a heat recovery section, HR-1, and a cooling section, R-1. The second effect consists of a brine heater H-2, a heat recovery section, HR-2, and a cooling section, R-2. The third effect consists of a brine heater H-3, a heat recovery section, HR-3, and a cooling section, R-3. The section between locations B and C serves a double purpose; it is the cooling section for the first effect, R-1, and the brine heater for the second effect H-2. Similarly, the section between locations E and F serves as the cooling section for the second effect, R-2, and the brine heater for the third effect, H-3. Sea water is used as a coolant in R-3.

F and L represent the flow rate of feed brine and flashing brine, respectively. The feed brine and recycle brine are referred to together as

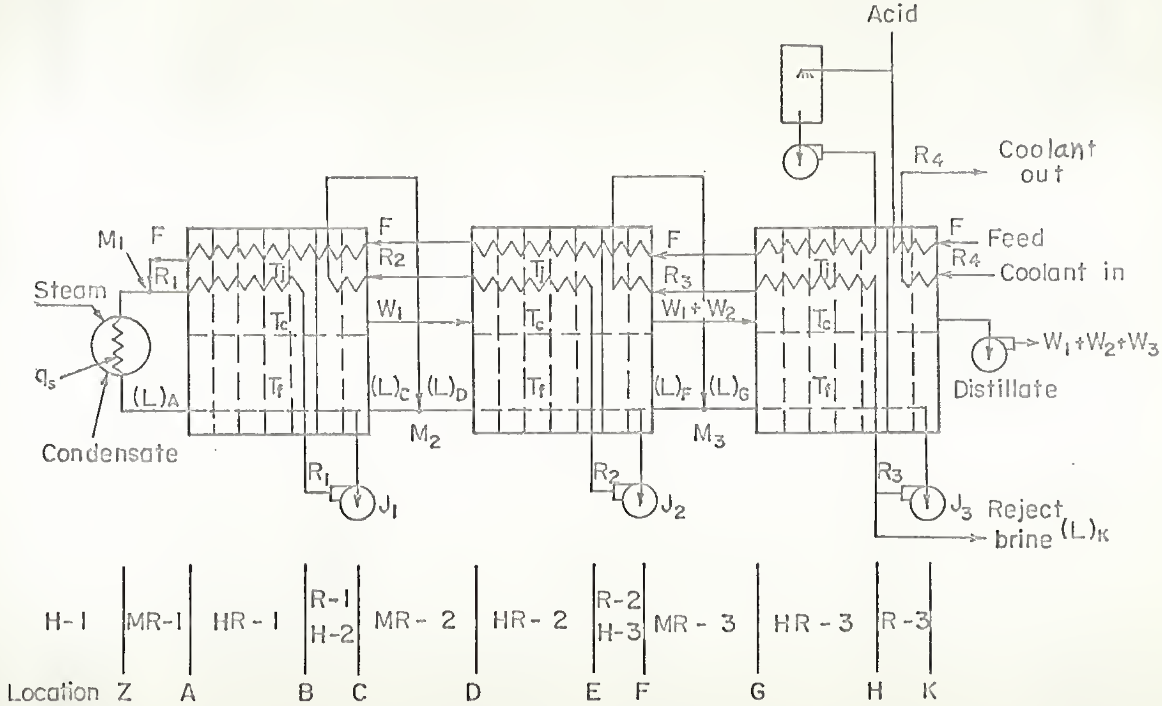

Fig. 5 Multi-effect multistage flash distillation system.

the non-flashing brine stream. T_f, T_j and T_c represent respectively the temperature of the flashing brine, non-flashing brine, and condensate, respectively. Subscript notation will be used to indicate the location. For example, $(T_f)_F$, $(T_j)_B$ and $(T_c)_H$ represent the temperature of the flashing brine at location F, temperature of the non-flashing brine at location B and the temperature of the condensate at location H respectively. R_1, R_2 and R_3 represent the recycle flow rate in the first, second and third effect, respectively, and W_1, W_2 and W_3 represent the condensate produced in the first, second, and third effects, respectively. R_4 represent the cooling water (sea water) used in the third effect.

The sea water feed is heated in R-3 and then acidified and degasified to remove CO_2 and other dissolved gases. After being heated successively in HR-3, H-3, HR-2, H-2, HR-1 it is mixed with recycle brine R_1 to form a brine stream which is heated in brine heater H-1 and then introduced into the first effect as the flashing brine $(L)_A$.

The flashing brine at location C is divided into two streams. One stream, $(L)_C$, is fed into the second effect and the other stream, R_1, is recirculated by a recycle pump, J_1, heated in HR-1 and then mixed with the feed stream at the mixing point M_1. As has been described, the combined stream is heated in brine heater H-1 and introduced into the first effect as flashing brine $(L)_A$. Similarly the flashing brine at locations F and K are divided into $(L)_F$ and R_2, and $(L)_K$ and R_3, respectively. $(L)_F$ is fed into the third effect, and $(L)_K$ is discharged from the third effect as the reject brine from the system. Streams R_2 and R_3 are recirculated by pumps J_2 and J_3 respectively, heated in sections HR-2 and H-2, and HR-3 and H-3, respectively, mixed with $(L)_C$ and $(L)_F$ at mixing points M_2 and M_3 respectively, and introduced to the second effect and the third effect as $(L)_D$ and

$(L)_G$, respectively.

The feed brine and the recycle brine are heated in each stage by the water vapor evaporated from the flashing brine in that stage. It is possible to arrange the flow-system so that the temperatures of the feed brine and the recycle brine are equal at any location. In the following discussion, such an arrangement is assumed. As has been described, the feed brine and the recycle brine are referred to jointly as the non-flashing brine and its temperature is denoted by T_j. The recycle brine, R_1, which is a part of the flashing brine at location C, is introduced into the condensing chamber at location B where it becomes a part of the non-flashing brine stream. Therefore, the following relation should hold.

$$(T_j)_B = (T_f)_C. \tag{28}$$

Similarly, we can write

$$(T_j)_E = (T_f)_F. \tag{29}$$

and

$$(T_j)_H = (T_f)_K. \tag{30}$$

Mixing is thermodynamically irreversible when two solutions, which differ in temperature and/or composition, are mixed together. The solutions mixed at the mixing points, M_1, M_2 and M_3, have differences in composition; however, by suitably locating point B, the temperature of the recycle solution R_2, $(T_j)_B$, can be adjusted to $(T_f)_C$, the temperature of stream $(L)_C$, and in this way the thermodynamic irreversibility due to mixing can be minimized. In this study isothermal mixing is at M_1, M_2, and M_3 and the heat of mixing of the sodium chloride-water system is neglected. Thus,

$$(T_j)_B = (T_f)_C = (T_f)_D. \tag{31}$$

Similarly,

$$(T_j)_E = (T_f)_F = (T_f)_G. \tag{32}$$

The unit enthalpy of dilute salt solutions is assumed to be a function of temperature but independent of composition. This assumption is justified because of the small heat of mixing and the rather limited concentration range of approximately from 3.5% to 7% encountered in this process.

A stage within each effect consists of a flashing chamber and a condenser chamber and a demister which separates the two chambers. When the flashing brine leaving one stage (the (n-1) th stage) is released into the next stage (the n-th stage), water vapor flashed out of the solution. The water vapor then passes through the demister on the way to the condenser chamber, where it is condensed to heat the nonflashing brine, i.e., the feed and recycle streams.

3.2 Explanation of the SUBROUTINE MODEL Used in the POP-II Program to Optimize the MEMS Plant

Fan, et. al. (7) have given the theoretical background and the derivation of the equations that are given in this section. The POP-II SUBROUTINE MODEL used in this example is listed in Table 2. The equations used in the SUBROUTINE MODEL are given immediately after the listing of SUBROUTINE MODEL in a one-to-one correspondence with the FORTRAN statements of SUBROUTINE MODEL. Also, some of the equation numbers are given beside the corresponding SUBROUTINE MODEL FORTRAN statements. For example, the first FORTRAN statement of SUBROUTINE MODEL under the PERFORMANCE EQUATIONS heading, is the same as Equation (33), the first equation in the following explanation of SUBROUTINE MODEL. It is hoped that this method of presentation

will give a clearer understanding of how a SUBROUTINE MODEL may be programmed.

The symbols used in SUBROUTINE MODEL are given in Table III-1 of Appendix III. Because of the simplicity of SUBROUTINE MODEL, no logic diagram is given for it.

Table 2. SUBROUTINE MODEL For The MEMS Plant Problem

```
      SUBROUTINE MODEL
      COMMON        P(5244)
      DIMENSION   X(50), Y(50), CONST(400)
      EQUIVALENCE (X(1),P(5120)),(Y(1),P(5180)),(CONST(1),P(451))
C     SIMULATION PROGRAM MEMS SEAWATER DISTILLATION PLANT
C     ****************************************************
C     CONSTANTS USED IN PERFORMANCE EQUATIONS
      AL=1000.
      B=17.01723
      AN1=23.
      AN2=23.
      AN3=22.
      CF=0.035
      CP=1.
      XA=230.
      XB=250.
      XC=270.
      XD=290.
      YA=958.8
      YB=945.5
      YC=931.8
      YD=917.5
      U0=510.
      U1=510.
      U2=510.
      U3=510.
C     ****************************************************
C     PERFORMANCE EQUATIONS
      Y(2)=X(1)+X(2)+X(3)                                    (33)
      Y(3)=Y(2)/(1.-CF/X(8))
      AL1O=Y(3)-X(1)                                         (35)
      AL2O=AL1O-X(2)
      AL3O=AL2O-X(3)
      AL1I=Y(3)+X(4)
      AL2I=Y(3)+X(5)-X(1)
      AL3I=Y(3)+X(6)-X(1)-X(2)
      CSF1O=CF*Y(3)/AL1O                                     (41)
      CSF2O=CF*Y(3)/AL2O
      CSF1I=CSF1O*(AL1O+X(4))/AL1I
      CSF2I=CSF2O*(AL2O+X(5))/AL2I
      CSF3I=X(8)*(AL3O+X(6))/AL3I
      Y(4)=X(10)-(AL/CP)*ALOG(CSF1O/CSF1I)                   (46)
      Y(5)=Y(4)-(AL/CP)*ALOG(CSF2O/CSF2I)
      Y(6)=Y(5)-(AL/CP)*ALOG(X(8)/CSF3I)
      A1=1.0100+(CSF1I+CSF1O)/(2.*0.0300)                    (49)
      A2=1.0075+(CSF2I+CSF2O)/(2.*0.0347)
      A3=0.3201+(CSF3I+X(8))/(2.*0.0315)
      Y(7)=Y(4)-A1                                           (52)
```

Table 2. (Con't)

```
      Y(8)=Y(5)-A2
      Y(9)=Y(6)-A3
      Z=X(9)
C     LAGRANGIAN POLYNOMIAL
      AA=YA*(Z-XB)*(Z-XC)*(Z-XD)/((XA-XB)*(XA-XC)*(XA-XD))
      AB=YB*(Z-XA)*(Z-XC)*(Z-XD)/((XB-XA)*(XB-XC)*(XB-XD))
      AC=YC*(Z-XA)*(Z-XB)*(Z-XD)/((XC-XA)*(XC-XB)*(XC-XD))
      AD=YD*(Z-XA)*(Z-XB)*(Z-XC)/((XD-XA)*(XD-XB)*(XD-XC))
      ALS=AA+AB+AC+AD
C     ******************************************************
      Y(10)=X(7)*ALS                                              (55)
      TLF0I=X(10)-Y(10)/(CP*(Y(3)+X(4)))                          (56)
      TLF1I=(CP*((AL10+X(5))*Y(4)+X(1)*Y(7))-Y(10))
     1        /(CP*(Y(3)+X(5)))
      TLF2I=(CP*((AL20+X(6))*Y(5)+(X(1)+X(2))*Y(8))-Y(10))
     1       /(CP*(Y(3)+X(6)))
      Y(15)=(Y(10)+Y(3)*CP*X(11)-(CP*(AL30*Y(6) +Y(2)*Y(9))))
     1        /(CP*(Y(6) -X(11)))                                 (59)
      DT0=X(9)-0.5*(X(10)+TLF0I)                                  (60)
      DT1=X(10)-TLF0I-A1-(X(10)-Y(4))/(2.*AN1)
      DT2=Y(4)-TLF1I-A2-(Y(4)-Y(5))/(2.*AN2)
      DT3=Y(5)-TLF2I-A3-(Y(5)-Y(6) )/(2.*AN3)
      Y(11)=Y(10)/(DT0*U0)                                        (64)
      Y(12)=X(1)*AL/(DT1*U1)
      Y(13)=X(2)*AL/(DT2*U2)
      Y(14)=X(3)*AL/(DT3*U3)
      HP1=X(4)*B*(EXP (-AL/(0.1104*(X(10)+460.)))                 (68)
     1    - EXP (-AL/(0.1104*(Y(4) +460.))))
      HP2=X(5)*B*(EXP (-AL/(0.1104*(Y(4) +460.)))
     1    - EXP (-AL/(0.1104*(Y(5) +460.))))
      HP3=X(6)*B*(EXP (-AL/(0.1104*( Y(5) +460.)))
     1    - EXP (-AL/(0.1104*(Y(6) +460.))))
C     ******************************************************
C     COST EQUATIONS
      Y(16)=CONST(1)*Y(3)/1.E+10                                  (71)
      Y(17)=CONST(2)*X(7)/1.E+10
      Y(18)=CONST(3)*HP1/1.E+10                                   (73)
      Y(19)=CONST(3)*HP2/1.E+10
      Y(20)=CONST(3)*HP3/1.E+10
      Y(21)=CONST(4)*Y(15)/1.E+10
      Y(22)=CONST(6)*Y(11)/1.E+10
      Y(23)=CONST(5)*Y(12)/1.E+10                                 (76)
      Y(24)=CONST(5)*Y(13)/1.E+10
      Y(25)=CONST(5)*Y(14)/1.E+10
      Y(1)=Y(16)+Y(17)+Y(18)+Y(19)+Y(20)+Y(21)
     1     +Y(22)+Y(23)+Y(24)+Y(25)+CONST(7)/1.E+10               (77)
      RETURN
```

POP-II PERFORMANCE EQUATIONS FOR A MEMS PLANT

The total production rate of fresh water, ΣW_n is equal to the sum of the individual distilled water streams from each effect

$$\Sigma W_n = W_1 + W_2 + W_3 \tag{33}$$

A salt material balance* around the entire plant gives the seawater feed rate

$$F = \frac{\Sigma W_n}{1 - \dfrac{C_o}{(Cf)_k}} \tag{34}$$

The flow rates of the effluent flashing brine streams for the individual effects are given, respectively, by

$$(L)_C = F - W_1 \tag{35}$$

$$(L)_F = (L)_C - W_2 \tag{36}$$

$$(L)_K = (L)_F - W_3 \tag{37}$$

The flow rates of the influent flashing brine streams for the individual effects are given, respectively, by

$$(L)_A = F + R_1 \tag{38}$$

$$(L)_D = F + R_2 - W_1 \tag{39}$$

$$(L)_G = F + R_3 - W_1 - W_2 \tag{40}$$

---

* In the following discussion, the concentration of a brine solution refers to the salinity as defined by Badger and Associates (8).

Salinity material balances give the concentrations of the flashing brine solutions leaving effects one and two respectively:

$$(C_f)_C = C_o F/(L)_C \tag{41}$$

$$(C_f)_F = C_o F/(L)_F \tag{42}$$

The concentrations for the flashing brine solutions entering effects one, two, and three, respectively, are given by the following material balance equations:

$$(C_f)_A = (C_f)_C \left((L)_C + R_1 \right)/(L)_A \tag{43}$$

$$(C_f)_D = (C_f)_F \left((L)_F + R_2 \right)/(L)_D \tag{44}$$

$$(C_f)_G = (C_f)_K \left((L)_K + R_3 \right)/(L)_G \tag{45}$$

As the flashing brine proceeds through the effect, it is cooled due to flashing. Fan, et. al. (7) derived an equation which gives the temperature drop of the flashing brine solution across an effect as a function of the ratio of the effect's outlet to inlet concentration. The equation can be rearranged to give the temperatures of the effluent flashing brine solutions for the first, second, and third effects respectively:

$$(T_f)_C = (T_f)_A - \lambda/C_p \ln\left((C_f)_C/(C_f)_A \right) \tag{46}$$

$$(T_f)_F = (T_f)_C - \lambda/C_p \ln\left((C_f)_F/(C_f)_D \right) \tag{47}$$

$$(T_f)_K = (T_f)_F - \lambda/C_p \ln\left((C_f)_K/(C_f)_G \right) \tag{48}$$

where λ is the latent heat of evaporation of steam in the effect and C_p is the heat capacity of the flashing brine.

The boiling point elevation in an effect is taken as a linear function of the average flashing brine concentration in the effect. Furthermore, it is assumed that a one degree Fahrenheit temperature drop occurs across the demister which is placed in each effect to prevent the flashing brine solution from splashing onto the overhead heat transfer area and into the distilled water collection troughs.

Badger and Associates (8) give graphs which can be linearized about a point to give the boiling point elevation in an effect as a linear function of the average flashing brine concentration.

Equations (49), (50), and (51) were formulated to calculate the combined effects of the average boiling point elevation and the demister temperature drop for effects one, two, and three respectively

$$(\alpha)_1,\text{av} = 1.0100 + \frac{1}{0.0300} \frac{((C_f)_A + (C_f)_C)}{2} \tag{49}$$

$$(\alpha)_2,\text{av} = 1.0075 + \frac{1}{0.0347} \frac{((C_f)_D + (C_f)_F)}{2} \tag{50}$$

$$(\alpha)_3,\text{av} = 0.3201 + \frac{1}{0.0315} \frac{((C_f)_G + (C_f)_K)}{2} \tag{51}$$

where $(\alpha)_n$,av is the combined effect of the average boiling point elevation and the demister temperature drop for the n-th effect.

The temperature of the distilled water leaving a stage is given by the flashing brine temperature leaving the stage minus the corresponding $(\alpha)_n$,av for that stage.

$$(T_c)_C = (T_f)_C - (\alpha)_1,\text{av} \tag{52}$$

$$(T_c)_F = (T_f)_F - (\alpha)_2,\text{av} \tag{53}$$

$$(T_c)_K = (T_f)_K - (\alpha)_3,\text{av} \tag{54}$$

The latent heat of vaporization of the saturated brine heater steam is written as a function of the steam temperature by using a Lagrangian Polynomial. However, in this study the brine heater steam temperature is allowed to vary only in one case study and then by only a small amount, because increasing the temperature will bring about a significant increase in the pressure within the brine heater. Higher pressures require that the brine heater be built out of sturdier, more expensive material. The effect of brine heater pressure upon brine heater cost has not been taken into account except at the brine heater temperature used for this study.

The rate of heat addition to the brine heater by the brine heater steam is given as

$$q_s = m_s \lambda_s \tag{55}$$

where m_s is the steam flow rate and λ_s is the latent heat of vaporization of the brine heater steam.

A heat balance around the brine heater, including all effects up to the (n + 1)th effect, will give the temperature of the seawater and recycle brine entering the n-th effect. Equations (56), (57), and (58) give the temperatures of the combined non-flashing brine streams entering the brine heater. effect one, and effect two, respectively. The seawater coming into the plant is assumed to be at an ambient seawater temperature of 85° F.

$$(T_j)_A = (T_f)_A - \frac{q_s}{C_p\,(F + R_1)} \tag{56}$$

$$(T_j)_C = \frac{C_p\,(((L_f)_C + R_2)\,(T_f)_C + W_1(T_c)_C) - q_s}{C_p\,(F + R_2)} \tag{57}$$

$$(T_j)_F = \frac{C_p\,(((L_f)_F + R_3)\,(T_f)_F + (W_1 + W_2)(T_c)_F - q_s}{C_p\,(F + R_3)} . \tag{58}$$

A heat balance around the entire plant will give the required cooling water flow rate

$$R_4 = \frac{q_s + FC_p\ (T_j)_K - C_p\ \left\{ (L)_K (T_f) + (\ \ W_n)\ (T_c)_K \right\}}{C_p\ \left\{ (T_f)_K - (T_j)_K \right\}} \tag{59}$$

The temperature driving force available for heat transfer in the brine heater is taken as the arithmetic mean of the approach temperature differences.

$$(\Delta t)_0 = T_s - (1/2)((T_f)_A + (T_j)_A) \tag{60}$$

Fan, et.al. (7) derive equations for calculating the effective heat transfer driving force for each effect. The equations take into account the boiling point elevation of the flashing brine, the demister temperature drop, and the fact that the effect has a finite number of stages rather than an infinite number. The effective heat transfer driving forces for effects one, two, and three are given by Equations (61), (62), and (63) respectively

$$(\Delta t)_1 = (T_f)_A - (T_j)_A - (\alpha)_{1,av} - (1/2)\frac{(T_f)_A - (T_f)_C}{N_1} \tag{61}$$

$$(\Delta t)_2 = (T_f)_D - (T_j)_D - (\alpha)_{2,av} - (1/2)\frac{(T_f)_D - (T_f)_F}{N_2} \tag{62}$$

$$(\Delta t)_3 = (T_f)_G - (T_j)_G - (\alpha)_{3,av} - (1/2)\frac{(T_f)_G - (T_f)_K}{N_3} \tag{63}$$

where, N_n is the number of stages in effect n.

The heat transfer areas required for the brine heater and each effect are given as follows

$$A_0 = q_s / (\Delta t)_0\ U_0 \tag{64}$$

$$A_1 = W_1 / (\Delta t)_1\ U_1 \tag{65}$$

$$A_2 = W_2 / (\Delta t)_2 U_2 \tag{66}$$

$$A_3 = W_3 / (\Delta t)_3 U_3 \tag{67}$$

where U_n is the overall heat transfer coefficient for the brine heater at n th effect. Equations (65), (66), and (67) give a somewhat approximate evaluation of the heat transfer areas of the heat rejecting sections of the respective effects. This approximation is thought to be justified since the sizes and thus the costs of the heat rejection areas are a small part of the total heat transfer area of each effect.

The power requirement of a recycle brine pump is taken to be proportional to the product of the pressure head which is developed by the pump and the recycle brine mass flow rate. The pressure head developed by the pump must be enough to overcome the frictional loss in the recirculation line and the pressure increase caused by different vapor pressures at the inlet and outlet of the stage. In order to determine the power requirements, the following assumptions are made:

1. The vapor pressure of the flashing brine solution is given by an integrated Clausius-Clapeyron equation.
2. The friction loss is proportional to the pressure drop between the outlet and inlet of the effect.
3. The density of the flashing brine solutions are constant throughout the plant.
4. The horsepower calculated has been corrected for the mechanical inefficiency of the pump.

The power requirements for effects one, two, and three, respectively, are given by

$$(HP)_1 = B_1 R_1 \left[\mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_A + 460.)}\right] - \mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_C + 460.)}\right]\right] \tag{68}$$

$$(HP)_2 = B_1 R_2 \left[\mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_C + 460.)}\right] - \mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_F + 460.)}\right]\right] \tag{69}$$

$$(HP)_3 = B_1 R_3 \left[\mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_F + 460.)}\right] - \mathrm{Exp}\left[\frac{-\lambda}{0.1104\,(\,(T_f)_K + 460.)}\right]\right] \tag{70}$$

where $(HP)_n$ is the power requirement for effect n and B_1 is a constant which takes into account the friction loss, the recirculating brine density, the mechanical efficiency of the pump, and the integration constant of the integrated Clausius-Clapeyron equation.

POP-II ECONOMIC EQUATIONS FOR A MEMS PLANT

Both initial equipment costs and operating costs for the life of the equipment are taken into account in the objective function. The following costs are considered to be significant in the selection of equipment size:

I. Initial equipment costs
   a. Heat transfer area costs
      1. Brine heater
      2. Each Effect
   b. Pumps - Brine recirculation
   c. Outer shell of each effect

II. Operating costs
   a. Feed brine pretreatment and pumping
   b. Cooling water pumping
   c. Recirculation brine pumping
   d. Brine heater steam

In the following cost equations, the cost coefficients have been divided by the total production rate of fresh water to obtain the total cost on the basis of a unit of production.

The initial cost of the brine feed pump, the brine pretreatment cost, and the brine feed pump operating cost are all considered to be proportional to the brine feed rate. Therefore,

$$E_1 = C_W F \tag{71}$$

where C_W is the unit feed water cost.

The brine heater steam cost is proportional to the steam consumption rate

$$E_2 = C_s m_s \tag{72}$$

where C_s is the unit steam cost.

The energy cost associated with the operation of the recycle pump in the n-th effect and the initial cost of each pump can be calculated by the following equation:

$$(C_e + \Psi C_J)\ (HP)_n \ = \ C_{HP}\ (HP)_n \qquad (73)$$

where C_e is the unit power cost, Ψ is the capitalization charge, C_J is the capital cost per horsepower, and $C_{HP} = (C_e + \Psi\ C_J)$.

The cooling water cost is directly proportional to the quantity used

$$E_3 \ = \ C_C\ R_4 \qquad (74)$$

where C_C is the unit cooling water cost.

The brine heater capital cost per hour, E_2, is given by

$$E_2 \ = \ \Psi C_B A_0 \ = \ C_{HT} A_0 \qquad (75)$$

where C_B is the capital cost per unit heat transfer area, and $C_{HT} = \Psi C_B$.

The heat transfer area cost for the n-th effect can be calculated as follows:

$$(E^n{}_3) \ = \ \Psi C_H A_n \ = \ C_{EF} A_n \qquad (76)$$

where A_n is the heat transfer area of the n-th effect, C_H is the capital cost per unit heat transfer area, and $C_{EF} = \Psi\ C_H$. There should be a different cost coefficient for the brine heater area cost than for the effect areas.

The total outer shell cost for the plant based on a unit of production is taken into account by adding a constant, E_6, the sum of the previous costs. The total water cost per unit of production is given by the sum of the cost terms previously stated.

The objective cost to be minimized is:

$$Y(1) = C_W F + C_s m_s + \sum_{n+1}^{N=3} C_{HP}(HP)_n + C_c R_4 + C_{HT} A_0 + \sum_{n=1}^{N=3} C_{EF} A_n + E_G \quad (77)$$

3.3 Results, Discussion and Conclusions Concerning the Usefulness of POP-II for Optimizing the MEMS Plant

The results of three representative runs made with the POP-II optimization program are summarized in Table 3. Runs 1, 2, and 3 were started at different feasible points. The initial and final feasible X(J), values and the total cost obtained for each run are given in the table. Three independent variables, the brine heater steam temperature, X(9), the temperature of the flashing brine entering effect one, X(10), and the temperature of the fresh seawater entering the plant, X(11), were held constant in this work so that the results could be compared with the work by Fan, et. al. (7). The complete computer output for Run 1, including the loop-to-loop output is given in Appendix II. The loop-to-loop outputs for Runs 2 and 3 are also given in Appendix II.

The starting values of X(J) used in Run 1 were essentially the same as the overall optimum for the MEMS process obtained by Fan, et. al. (7) using the discrete version of the maximum principle. The total cost for this set of initial input values was $0.2867 per thousand gallons of potable water, which is again very close to the optimal value obtained by Fan, et. al. As shown in Table 3, an appreciable change occurred between the initial and final values of only two independent variables, $X(7)= m_s$ and $X(8)=(C_f)_k$. The brine heater steam consumption rate, m_s, decreased by

1.9 percent. The concentration of the discharge brine solution, $(C_f)_k$, increased by 14.1 percent. The final total cost was \$0.2866 per thousand gallons of potable water produced which is again very close but slightly higher than that obtained by Fan, et. al. Since Run 1 was the lowest cost case obtained with POP, the final values of the dependent variables are given in Table 4.

The total cost essentially stayed constant throughout Run 1. No infeasible intermediate loops were encountered during the run as can be seen by noting that the production equality constraint was always satisfied and the independent and dependent variables remained positive throughout the run. The adaptive move limit option was used in this run. The run terminated at the end of twenty-five loops because the calculated move limits became too small.

Run 2 was started with all X(J) values except X(3) less than the corresponding starting X(J) values of Run 1. The initial total cost was \$0.2926 per thousand gallons of potable water produced. As was the case in Run 1, only the brine heater steam consumption rate and the concentration of the discharge brine solution changed appreciably during the run. The brine heater steam consumption rate increased by 10.8 percent. The concentration of the discharge brine solution increased by 14.6 percent. The final total cost, \$0.2890 per thousand gallons of potable water produced, was only 0.84 percent larger than the optimum cost found in Run 1.

During the course of Run 2, several loops ended at infeasible points. For these loops, the process required a negative amount of cooling water in effect three, R_4. The next loop after an infeasible loop was always feasible and the last ten loops were all feasible. Also, small oscillations in the total cost in going from one loop to the next were noted at the beginning

Table 3. Initial and Final Feasible Values of Independent Variables for Runs 1, 2, and 3

Run 1

	Initial Value	Final Value
Effect 1 Distillate Production (lbs./hr.), $W_1 = X(1)$	2,602.3200	2,602.3200
Effect 2 Distillate Production (lbs./hr.), $W_2 = X(2)$	2,829.8900	2,829.8600
Effect 3 Distillate Production (lbs./hr.), $W_3 = X(3)$	2,907.7900	2,907.7900
Effect 1 Recycle Rate (lbs./hr.), $R_1 = X(4)$	39,800.0000	39,800.0000
Effect 2 Recycle Rate (lbs./hr.), $R_2 = X(5)$	42,400.0000	42,400.0000
Effect 3 Recycle Rate (lbs./hr.), $R_3 = X(6)$	46,000.0000	46,000.0000
Feed Rate Brine Heater Steam (lbs./hr.), $m_s = X(7)$	531.0000	521.0000
Effect 3 Discharge Brine Concentration (wt. %), $(C_f)_K = X(8)$	0.0640	0.0649
Total Cost	0.2867	0.2866

Table 3. (Continued)

Run 2

	Initial Value	Final Value
Effect 1 Distillate Production (lbs./hr.), $W_1 = X(1)$	2,231.1640	2,231.164
Effect 2 Distillate Prouuction (lbs./hr.), $W_2 = X(2)$	2,426.2760	2,426.2760
Effect 3 Distillate Production (lbs./hr.), $W_3 = X(3)$	3,682.5590	3,682.5301
Effect 1 Recycle Rate (lbs./hr.), $R_1 = X(4)$	34,123.5195	34,123.5195
Effect 2 Recycle Rate (lbs./hr.), $R_2 = X(5)$	36,352.6997	36,352.6997
Effect 3 Reeycle Rate (lbs./hr.), $R_3 = X(6)$	39,439.2500	39,439.2500
Feed Rate Brine Heater Steam (lbs./hr.), $m_s = X(7)$	455.2660	510.2660
Effect 3 Discharge Brine Concentration (wt. %), $(C_f)_K = X(8)$	0.0540	0.0548
Total Cost	0.2926	0.2890

Table 3. (Continued)

Run 3

	Initial Value	Final Value *
Effect 1 Distillate Production (lbs./hr.), $W_1 = X(1)$	3,903.5000	2,068.099
Effect 2 Distillate Production (lbs./hr.), $W_2 = X(2)$	4,244.8000	2,270.085
Effect 3 Distillate Production (lbs./hr.), $W_3 = X(3)$	191.70000	4,001.500
Effect 1 Recycle Rate (lbs./hr.), $R_1 = X(4)$	59,700.0000	54,269.864
Effect 2 Recycle Rate (lbs./hr.), $R_2 = X(5)$	63,600.0000	35,727.872
Effect 3 Recycle Rate (lbs./hr.), $R_3 = X(6)$	69,000.0000	62,814.294
Feed Rate Brine Heater Steam (lbs./hr.), $m_s = X(7)$	796.5000	570.409
Effect 3 Discharge Brine Concentration (wt. %), $(C_f)_K = X(8)$	0.0960	0.079
Total Cost	0.3340	0.306

*Final feasible value was at the end of loop 6.

Table 4. Optimal Values for the MEMS Plant Calculated by POP-II

Mass Flow Rates (lbs./hr.):			W_n	=	8,339.97	$(T_f)_C$	=	204.26
F	=	18,144.10				$(T_f)_F$	=	154.42
$(L)_A$	=	57,902.50	Concentrations (wt. %):			$(T_f)_K$	=	103.85
$(L)_C$	=	57,900.20	$(C_f)_A$	=	0.0390	$(T_j)_A$	=	241.64
$(L)_D$	=	58,670.30	$(C_f)_C$	=	0.0409	$(T_j)_C$	=	195.92
$(L)_F$	=	15,500.20	$(C_f)_D$	=	0.0476	$(T_j)_F$	=	146.15
$(L)_G$	=	12,670.30	$(C_f)_F$	=	0.0500	$(T_j)_K$	=	84.99
$(L)_K$	=	9,762.54	$(C_f)_G$	=	0.0617	T_s	=	274.39
m_s	=	521.00	$(C_f)_K$	=	0.0649	$(\Delta t)_0$	=	28.58
R_1	=	39,800.00	C_o	=	0.0350	$(\Delta t)_1$	=	5.01
R_2	=	42,400.00				$(\Delta t)_2$	=	4.59
R_3	=	46,000.00	Temperatures (^{o}F):			$(\Delta t)_3$	=	4.27
R_4	=	9,521.88	$(Tc)_C$	=	201.92	$(\alpha)_{1,av}$	=	2.34
W_1	=	2,602.32	$(Tc)_F$	=	152.01	$(\alpha)_{2,av}$	=	2.41
W_2	=	2,829.86	$(Tc)_K$	=	101.55	$(\alpha)_{3,av}$	=	2.33
W_3	=	2,907.79	$(T_f)_A$	=	250.10			

Table 4 (Continued)

Pump Motor Sizes (H.P.):

$(HP)_1$	=	1.14
$(HP)_2$	=	0.58
$(HP)_3$	=	0.22

Heat Transfer Areas (ft^2)

A_0	=	33.32
A_1	=	1,017.75
A_2	=	1,207.70
A_3	=	1,335.87

Costs (\$/1000 gal.)

A_0	=	0.0013
A_1	=	0.0244
A_2	=	0.0289
A_3	=	0.0320
F	=	0.0320
M_s	=	0.1302
R_1	=	0.0066
R_2	=	0.0033
R_3	=	0.0013
R_4	=	0.0057
Total	=	0.2866

of the run. This run also terminated at the end of twenty-five loops because the calculated move limits became too small.

Run 3 was started with all X(J) values except X(3) greater than the corresponding X(J) values of Run 1. The initial cost was \$0.3340 per thousand gallons of potable water produced. In this run all of the independent variables changed appreciably during the run. The significant changes in the values of the independent variables can perhaps be explained by the fact that the error limits on some of the dependent variables were larger in this run than in Runs 1 and 2. Furthermore, the initial move limits of the independent variables were larger in this run than in Runs 1 and 2. Loop number six was the best loop obtained in this run. The total cost at the end of loop six was \$0.306 per thousand gallons of potable water produced. Even though this run ended infeasible, the total cost at the end of loop six was only 6.8 percent higher than the optimum cost obtained in Run 1.

At the end of loop one of Run 3, the total cost was nearly three times the starting total cost. Thereafter, the total cost progressively decreased until the end of loop six. The infeasible loops of this run had negative cooling water requirements for effect three. This run terminated at the end of eleven loops because the calculated move limits became too small.

Preliminary simulation of the operation of the MEMS plant was carried out using computer program MODEL. The main portion of MODEL consists of the FORTRAN statements used in SUBROUTINE MODEL. MODEL's main function was to debug the FORTRAN statements used in SUBROUTINE MODEL. Since remote computer facilities were used to run the POP program, it was decided to debug SUBROUTINE MODEL's FORTRAN statements on available computer facilities

in order to speed up the debugging process. Normally, the POP program may be used to do the debugging. The starting points of Runs 2 and 3 were also determined by using MODEL. MODEL is explained in Appendix III.

The loop-to-loop output is a very useful aid for studying a particular SUBROUTINE MODEL. Three general observations about the MEMS SUBROUTINE MODEL were made by inspecting the loop-to-loop outputs for the three runs. First, there appeared to be a strong interaction between the brine heater steam consumption rate, m_s, and the discharge brine concentration, $(C_f)_k$. This was readily seen by observing the loop-to-loop output of Run 2. Secondly, small charges in m_s and $(C_f)_k$ in m_s and $(C_f)_k$ caused rather large changes in the cooling water requirement, R_4. Most of the runs which became infeasible did so because R_4 became negative. Many times R_4 became negative after only a small change in either m_s or $(C_f)_k$.

The DYDX matrix is also very useful for studying a particular SUBROUTINE MODEL. For the MEMS SUBROUTINE MODEL most of the a_{ij} elements of the m_s and $(C_f)_k$ columns of the DYDX matrices of all three runs were large in comparison to the a_{ij} elements of the other independent variables. Consequently, as was observed in the loop-to-loop outputs, small changes in these independent variables caused relatively large changes in the other dependent variables in addition to the total cost. Most of the a_{ij} elements of the cooling water, R_4, row of the DYDX matrices were larger than the elements of the other rows. This explains why small changes in most of the independent variables had a large effect on the cooling water requirements.

Several conclusions regarding the usefulness of POP have been formed in working with the MEMS plant problem. It is not a routine matter to apply POP to optimize complex processes. Experienced programmers may even

be required to use POP on simple models. It appears that some experimentation with the modes of operation and the input data may be required on simple models.

In the next section, future work is proposed in which changes in the mode of operation and the input data cards can be made now that more information is available on the SUBROUTINE MODEL used in this work.

3.4 Suggestions for Further Work

There are several possible changes that can be made in the POP input data forms and in the manner of operating POP which may help POP to converge to the optimum when it is started at points which are far away from the optimum.

A minimum limit can be put on the dependent variable R_4 which hopefully will prevent it from becoming negative. The optimizer may be run without using the adaptive move limit calculation feature. This mode of operation should enable the optimizer to get closer to the minimum cost during a run. Since the model is sensitive to small changes in the independent variable $(C_f)_k$, this variable can be held at a different constant value for a series of runs as was done by Fan et. al. (7). Furthermore, a reformulation of SUBROUTINE MODEL using different independent variables may be advisable.

Oscillations in the total cost may possibly be reduced by making the following adjustments: Move limits on the less sensitive independent variables may be increased. It may be necessary to individually adjust each move limit during a series of runs. In combination with the above steps, changes in the values of the constants used in the optimizer's convergence logic may help.

A general procedure for conducting POP runs is outlined next. Analyze the loop-to-loop output and the DYDX matrix. Make necessary adjustments such

as those previously described. Restart the next run from the best point obtained in the previous run. Many runs which do not end at a feasible point contain useful information and often one of their loops is an excellent starting point for the next run. See loop six of Run 3. Sample input data forms, which contain the above changes are given in Appendix I.

A more elaborate process simulation program or SUBROUTINE MODEL can be investigated in the future. The number of stages per effect can easily be made an independent variable as is suggested by Fan, et. al. (7). Information giving equipment cost as a function of equipment size and steam cost as a function of the steam's physical properties (9) is becoming more readily available as time progresses.

Certain advantages may be realized by combining POP with other optimization techniques such as the discrete maximum principle.

4.0 REFERENCES

1. Hadley, G., "Nonlinear and Dynamic Programming," Addison-Wesley, Reading, Mass. (1964).

2. Kuhn, H. W., and Tucker, A. W., "Nonlinear Programming", Proceedings of the Second Berkley Symposium on Mathematical Statistics and Probability, J. Neyman, Ed., U. of California Press, Berkeley, Calif. (1951).

3. Carr, C. R., and C. W. Howe, "Quantitative Decision Procedures in Management and Economics", McGraw-Hill, New York (1964).

4. Wilds, D. J., and C. S. Beightler, "Optimization Theory", Prentice-Hall, Englewood Cliffs, New Jersey (1964).

5. Smith, H. V., "A Process Optimization Program for Nonlinear Systems: POP-II", POP-II 7090 H9IBM0021, IBM Share General Program Library, IBM Corp., Houston, Texas 77025.

6. Baumol, W. J., "Economic Theory and Operations Analysis", Prentice-Hall, Englewood Cliffs, N. J. (1961).

7. Fan, L. T., and Associates, "Analysis and Optimization of a Multi-effect, Multistage Flash Evaporation System," Special Report No. 74, Kansas Engineering Experiment Station, Manhattan, Kansas.

8. Badger, W. L. and Associates, "Critical Review of Literature on Formation and Prevention of Scale", Office of Saline Water Research and Development Progress Report No. 25, U. S. Department of the Interior, Washington 25, D.C. (1959).

9. Detman, R. F., Chem. Eng. Progr. 63, 80-89 (1967).

5.0 NOMENCLATURE

a_i	A constant in several equations.
a_{ij}	As used in equations (15) and (17), a constant which denotes the coefficient of the i th row and the j th column of a system of linear equations. As used in equation (26) it represents the partial derivative of the i th equality constraint with respect to the j th independent variable.
A_0	Heat transfer area of the brine heater, sq. ft.
A_1	Heat transfer area of the first effect, sq. ft.
A_2	Heat transfer area of the second effect, sq. ft.
A_3	Heat transfer area of the third effect, sq. ft.
b_i	A constant which is first introduced in the equality constraint, equation (1b).
B_1	A constant used in the equations which calculate the horsepower requirements of the recycle pumps, H.P.-hr./lb.
C_e	Unit power cost, $/H.P.
$(C_f)_A$	Concentration of the flashing brine stream flowing into the first effect, wt. %.
$(C_f)_C$	Concentration of the flashing brine stream flowing out of the first effect, wt. %.

$(C_f)_D$	Concentration of the flashing brine stream flowing into the second effect, wt. %.
$(C_f)_F$	Concentration of the flashing brine stream flowing from the second effect, wt. %.
$(C_f)_G$	Concentration of the flashing brine stream flowing into the third effect, wt. %.
$(C_f)_K$	Concentration of the flashing brine stream flowing from the third effect, wt. %.
C_{ij}	The coefficient of the quadratic term of equation (16)
C_o	Concentration of seawater fed to the MEMS plant, wt. %.
C_p	Heat capacity of flashing brine solution, Btu/lb. ^oF.
C_S	Unit steam cost, \$/lb.
C_B	Unit cost of brine heater heat transfer area, \$/sq. ft.
C_C	Unit cooling water cost, \$/lb.
C_{EF}	Equal to ψC_H
C_H	Unit cost of heat transfer area in an effect, \$/sq. ft.
C_{HP}	Equal to $(C_e + \psi C_J)$
C_{AT}	Equal to ψC_B
C_J	Capital cost per horsepower, \$/H.P.
C_W	Unit feed water cost, \$/lb.

CONST(I)	The i th constant in the array of constants which can be used with POP-II
DYDX	The name of the matrix printed out at the end of a POP-II run which can be used to make an incremental sensativity analysis near the point at which the run ends.
e_i	The value of the i th equality constraint when it evaluated at a feasible point
E_1	Total cost associated with the seawater fed to the MEMS plant, \$/1000 gal. fresh water
E_2	Total cost of brine heater steam, \$/1000 gal. fresh water
E_3	Total cost of cooling water, \$/1000 gal. fresh water
(E_3^n)	Total cost of heat transfer area for the n th effect \$/1000 gal. fresh water
E_6	Total cost of the outer shell of the MEMS plant based on 1000 gal. fresh water produced, \$/1000 gal. fresh water
f(x)	The objective function
F	Feed rate to the MEMS plant, lbs/hr.
$g_i(x)$	The i th equality or inequality constraint
h(x)	A function used in the discussion about the quadratic form
$h_i(x)$	The i th equality constraint (used in the explanation of POP-II)

$(HP)_1$	Horsepower required for the recycle pump of the first effect, H.P.
$(HP)_2$	Horsepower required for the recycle pump of the second effect, H.P.
$(HP)_3$	Horsepower required for the recycle pump of the third effect, H.P.
L	The Lagrangian function used in the development of the Kuhn-Tucker conditions.
$(L)_A$	Flow rate of lashing brine stream into the first effect, lbs./hr.
$(L)_C$	Flow rate of lashing brine stream from the first effect, lbs./hr.
$(L)_D$	Flow rate of lashing brine stream into the second effect, lbs./hr.
$(L)_F$	Flow rate of flashing brine stream from the second effect, lbs./hr.
$(L)_G$	Flow rate of flashing brine stream into the third effect, lbs./hr.
$(L)_K$	Flow rate of flashing brine stream from the third effect, lbs/hr.
m_s	Brine heater steam flow rate, lbs/hr
q_s	Heat transfer rate in the brine heater, BTU/hr.
R_1	Recycle brine flow rate in the first effect, lbs./hr.

R_2	Recycle brine flow rate in the second effect, lbs./hr.
R_3	Recycle brine flow rate in the third effect, lbs./hr.
R_4	Cooling water flow rate in the third effect, lbs./hr.
S	Value of the objective function of an optimization problem
$(Tc)_C$	Temperature of distillate water from effect one, ^oF
$(Tc)_F$	Temperature of distillate water from effect two, ^oF
$(Tc)_K$	Temperature of distillate water from effect three, ^oF
$(T_f)_A$	Temperature of flashing brine stream into effect one, ^oF
$(T_f)_C$	Temperature of flashing brine stream from effect one, ^oF
$(T_f)_F$	Temperature of flashing brine stream from two, ^oF
$(T_f)_K$	Temperature of flashing brine stream from effect three, ^oF
$(T_j)_A$	Temperature of combined recycle and feed streams into the brine heater, ^oF
$(T_j)_C$	Temperature of combined recycle and feed streams into effect one, ^oF

$(T_j)_F$	Temperature of combined recycle and feed streams into effect two, ^oF
$(T_j)_K$	Temperature of combined recycle and feed streams into effect three, ^oF
T_s	Temperature of brine heater steam, ^oF
$(\Delta t)_0$	Temperature differences used for heat transfer in brine heater, ^oF
$(\Delta t)_1$	Temperature difference used for heat transfer in effect one, ^oF
$(\Delta t)_2$	Temperature difference used for heat transfer in effect two, ^oF
$(\Delta t)_3$	Temperature difference used for heat transfer in effect three, ^oF
u_i	Slack variable used in development of Kuhn-Tucker Conditions
U_0	Overall heat transfer coefficient for the brine heater, Btu/hr-ft^2-^oF
U_1	Overall heat transfer coefficient for effect one, Btu/hr-ft^2-^oF
U_2	Overall heat transfer coefficient for effect two, Btu/hr-ft^2-^oF
U_3	Overall heat transfer coefficient for effect three, Btu/hr-ft^2-^oF
$U1$	Equal to U_1
W_1	Rate of distillate water production in effect one, lbs./hr.

W_2	Rate of distillate water production in effect two, lbs./hr.
W_3	Rate of distillate water production in effect three, lbs./hr.
$\sum W_n$	Total rate of distillate water production in the MEMS plant, lbs./hr.
x	An abbreviated notation which stands for the point $(x_1, x_2, \cdots, x_s)$ in s-space.
x_0	A feasible point $(x_{1_0}, x_{2_0}, \cdots, x_{s_0})$
X(I)	The i th independent variable (used in POP programming)
y_i	The i th dependent variable
$y_i(x_0 + \delta x)$	The i th dependent variable approximated as a linear function of δx
Y(I)	The i th dependent variable (used in POP programming)
Y(1)	The objective function (used in POP programming)

Greek Letters

$(\alpha)_{n,av}$	The average boiling point elevation in the n th effect
λ	Latent heat of vaporization of flashing brine stream in an effect, Btu/lb.
λ_i	Lagrange multiplier
λ_s	Latent heat of steam used in brine heater, Btu/lb.
ψ	Capitalization charge

δx_j	Distance measured in j th direction from x_0

APPENDIX I

INPUT DATA FORMS FOR PROPOSED FUTURE WORK ON THE ALAS PROCESS

The data forms are generally self-explanitory; however, a detailed explanation of the data required by POP is given by Smith (1).

POP II

TITLE (72 CHARACTERS)	PROG. NO.	PAGE	SEQ.
3 EFFECT MEMS SEAWATER DISTILLATION PLANT		01	01
1	73	77	79

NO. OF INDEPENDENT VARIABLES	NO. OF DEPENDENT VARIABLES	NO. OF CONSTANTS		PROG. NO.	PAGE	SEQ.
11	25	7			01	02
1	6	11	16	73	77	79

FORM 2
KW1 -20

POP II

Columns	Value	Description
1–3	0	CHANGE CASE OPTION (1 = BASE 0 = CHANGE CARDS)
4–6	5 0	MAXIMUM NO. OF. OPTIMIZATION LOOPS
7–9	0	USE SOLUTION VALUE OF INDEPENDENT AND DEPENDENT VARIABLES AS INPUT TO NEXT CASE (1 = ON 0 = OFF)
10–12	0	NO. OF LOOPS TO PRINT DERIVATIVES MATRIX
13–15	0	PRINT DETAILS OF ADAPTIVE MOVE LIMIT CALCULATIONS AND LP VALUE OF DEPENDENT VARIABLES (1 = ON 0 = OFF)
16–18	0	DUMP CONTROL: – = CONDENSED OUTPUT PER LP ITERATION, 0 = NO OUTPUT, 1 = BEFORE AND AFTER, 2 = EACH SOLUTION
19–21	1	RUN NUMBER
22–24	1	MONTH
25–27	3 0	DAY
28–30	0 7	YEAR
31–33	0	NO. OF LOOPS TO PRINT EXTRA DEBUGGING OUTPUT FROM WRITE (2) AND OUTPUT SUBROUTINES
34–36	0	PRINT CORRECTED INPUT DATA FOR EACH CASE FROM SUBROUTINE WRITE (1) AND (2)
37–39	0	VARIABLES OF PREVIOUS LOOP RESTORED PRIOR TO CURRENT LOOP IF PROFIT INCREASE NEGATIVE ON PRIOR LOOP (1 = ON 0 = OFF)
40–42	0	NO. OF LOOPS WITH ON-LINE TIMING PRINTOUT
43–45	0	PRINT X'S TO EIGHT SIGNIFICANT FIGURES AT END OF EACH LOOP (1 = ON 0 = OFF)
46–48	2 0	NO. OF CONSECUTIVE LARGE MOVES BEFORE INCREASING MOVE LIMIT MULTIPLIER
49–51	1	RUN TIMING VIA CORE CLOCK (1 = YES 0 = NO)
52–54	1	TAKE PANIC EXIT DURING PARTIALS CALCULATION IF ANY X IS DECREMENTED TO ZERO OR NEGATIVE (1 = ON 0 = OFF)
55–57	1	ALLOW TEMPORARY DOUBLING OF MOVE LIMIT (1 = ON 0 = OFF)
58–60	0	NO. OF LOOPS WITH LP MATRIX DUMPS

PROG. NO.
73–80 [| 2 | 01]

FORM 3
KW 21–40

POP II

1–3	0	NOT USED EXCEPT WITH 7040
4–6	1	RESERVED FOR CHAIN LINK OR OVERLAY VERSION OF POP
7–9	0	
10–12	0	CALCULATE ADAPTIVE MOVE LIMITS IN SUBROUTINE PARTL (1 = ON, 0 = OFF)
13–15	0	BASIS CALCULATION ONLY FOR THIS RUN (1 = ON 0 = OFF)
16–18	0	NOT USED
19–21	0	NOT USED
22–24	0	NOT USED
25–27	2 0	NO. OF CONSECUTIVE SMALL MOVES PERMITTED BEFORE DECREASING MOVE LIMIT MULTIPLIER
28–30	0	WRITE 1 LINE SUMMARY OF EACH LOOP (1 = ON 0 = OFF)
31–33	0	NOT USED
34–36	0	NOT USED
37–39	0	NOT USED
40–42	0	NOT USED
43–45	0	NOT USED
46–48	0	NOT USED
49–51	0	NOT USED
52–54	1	TYPE OF OBJECTIVE FUNCTION, Y (1) (1 = MINIMIZATION, 0 = MAXIMIZATION)
55–57	0	Y (2) MIN. AND MAX. LIMITS SET EQUAL TO PRODUCTION CALCULATED BY SUBROUTINE BASIS
58–60	0	NOT USED

PROG. NO.

73–80 | | 05 | 01 |

POP II
PROGRAM CONSTANTS
(SUGGESTED VALUES)

Column	Value	Description
1	.00001	ESSENTIAL ZERO FOR LP MATRIX ELEMENTS
11	.00001	ESSENTIAL ZERO FOR FUNCTIONAL ELEMENTS
21	500.0	ARTIFICIAL HIGH COST FOR FORCE VECTOR
31	.01	ESSENTIAL ZERO FOR FRACTIONAL MOVE TEST
41	.05	FACTOR USED TO DETERMINE IF CURRENT SOLUTION IS OPTIMAL
51	.005	DEAD ZONE FRACTION FOR LIMIT LABELS ON SUMARY OUTPUT

CARD NO.

73	77	79
	04	01

POP II
PROGRAM CONSTANTS (CONT.)
(SUGGESTED VALUES)

Field (column)	Value	Description
1	1.0	INITIAL VALUE OF MOVE LIMIT MULTIPLIER
11	2.0	MULTIPLIER USED IN ADAPTIVE MOVE LIMIT CALCULATIONS (OPTIONAL)
21		NOT USED
31	1.0	MULTIPLIER USED WHEN KW (39) $\neq$ 0
41	.005	FACTOR USED TO DECIDE IF IMPROVEMENT FROM OPTIMIZER LOOP IS SMALL OR LARGE
51	.5	FACTOR USED AS MULTIPLIER OR DIVISOR IN INCREASING OR DECREASING MOVE LIMIT MULT.

PROG. NO.

73	77	79
	03	01

POP II
MODEL CONSTANTS

1	11	21	31	41	51		PROG. NO. 73	PG. 77	SEQ. 79
17650.	2500000.	57590000.	5984.	239700.	370000.			06	01
[illegible]05000000.								06	02
									03
									04
									05
									06
									07
									08
									09
									10
									11
									12
									13
									14
									15
									16
									17
									18
									19
									20

POP II
INDEPENDENT VARIABLES

TITLE	STARTING VALUE	MIN LIMIT	MAX LIMIT	DELTA	MOVE LIMIT	MOVE PENALTY	PROG. NO.	PAGE	SEQ.
W1	3903.5	1.1	90000.0	1.0	200.0			07	01
W2	4244.8								02
W3	191.7								03
R1	59700.0				1000.0				04
R2	63600.0								05
R3	69000.0								06
MS	796.5				10.0				07
CSF30	0.096	0.01	1.0	0.0005	0.005				08
PS	274.4	274.41	274.41						09
TSF11	250.0	250.0	250.01						10
TLF31	85.0	85.0	85.01						11
1	21	29	37	45	53	61	73	77	79

POP II
DEPENDENT VARIABLES *

TITLE		STARTING VALUE	MIN LIMIT	MAX LIMIT	DELTA	MAX. Y ERROR PER LOOP			PROG. NO.	PG.	SEQ.
TOTAL COST			0.01		1.0	0.01				08	01
PRODUCTION			8339.5	8340.5		1.00					02
FEED			0.01			1.00					03
TSF10						0.5					04
TSF20						0.5					05
TSF30						0.5					06
TC10						0.5					07
TC20						0.5					08
TC30						0.5					09
[illegible]						10.					10
AREA 0						0.5					11
AREA 1						1.0					12
AREA 2						1.0					13
AREA 3						1.0					14
COOL W						1.0					15
1	19	21	29	37	45	53	61	69	73	77	79

*NOTE: STARTING VALUE IS OPTIONAL.

POP II
DEPENDENT VARIABLES *

TITLE		STARTING VALUE	MIN LIMIT	MAX LIMIT	DELTA	MAX. Y ERROR PER LOOP			PROG. NO.	PG.	SEQ.
F COST					1.0	0.01				03	16
MS COST											17
R1 COST											18
R2 COST											19
R3 COST											20
CW COST											21
AO COST											22
A1 COST											23
A2 COST											24
A3 Cost											25
1	19	21	29	37	45	53	61	69	73	77	79

NOTE: STARTING VALUE IS OPTIONAL

POP II
CONTROL CARDS

CONTROL WORD	ARRAY NAME	SUB-SCRIPT	VALUE	ARRAY NAME	SUB-SCRIPT	VALUE	ARRAY NAME	SUB-SCRIPT	VALUE		PROG. NO.	PAGE	SEQ.
COMPUT												09	01
CHANGE	I	24	1										02
COMPUT													03
CHANGE	X	1	2002.32	X	2	2829.80	X	3	2907.79				04
CHANGE	X	4	39800.0	X	5	42400.0	X	6	40000.0				05
CHANGE	X	7	521.0	X	8	0.0049	X	9	274.4				06
CHANGE	X	10	250.	X	11	85.0	KW	3	1				07
CHANGE	KW	24	0										08
COMPUT													09
CHANGE	KW	24	1	KW	3	0							10
COMPUT													11
CHANGE	KW	13	1										12
COMPUT													13
FINISH													14
1	7	13	17	25	31	35	43	49	53	61	73	77	79

ARRAY NAMES, SUBSCRIPTS, AND VALUES ARE TO BE USED ON CHANGE CARDS ONLY

LEFT JUSTIFY

REFERENCE

1. Smith, H. V., "A Process Optimization Program for Nonlinear Systems: POP-II," POP-II 7090 H9IBM0021, IBM Share General Program Library, IBM Corp., Houston, Texas 77025.

APPENDIX II

POP-II SAMPLE OUTPUT

The computer output for Run 1 is given in this Appendix. Run 1 is the optimum case for this work. Also, the loop-to-loop computer outputs for Runs 2 and 3 are given in this Appendix.

```
RUN NUMBER  1       PROBLEM NUMBER  3       LOOP NUMBER  0       12/12/66
INPUT DATA FOR PLANT OPTIMIZATION PROGRAM, PAGE   1

*****  KW - KEY WORDS  ****************
  -0   50  -0  -0  -0  -0    3   12   12   66  -0    1  -0  -0  -0    5    1    1    1  -0
  -0    1  -0    1  -0  -0  -0  -0    5  -0  -0  -0  -0  -0  -0  -0  -0    1  -0  -0

***  CST - OPTIMIZER CONSTANTS ********
   0.099999994E-04   0.099999994E-04   0.500000000E 03   0.999999985E-02   0.499999993E-01   0.499999993E-02
  -0.                0.199999996E 01  -0.                -0.                0.500000000E 00   0.500000000E 00

******  CONST - MODEL  COEFFICIENTS****
   0.176499993E 05   0.250000000E 07   0.575899996E 08   0.598399997E 04   0.239699997E 06   0.375999995E 06
   0.207999997E 09
```

RUN NUMBER 1   PROBLEM NUMBER 3   LOOP NUMBER 0   12/12/66

INPUT DATA FOR PLANT OPTIMIZATION PROGRAM, PAGE 2

***** INDEPENDENT VARIABLES *****

VARIABLE NAME	INPUT	MIN. LIMIT	MAX. LIMIT	DELTA	MOVE LIMIT	MOVE PENALTY
W1	0.26023E 04	-0.	0.90000E 05	0.10000E 02	0.10000E 03	-0.
W2	0.28299E 04	-0.	0.90000E 05	0.10000E 02	0.10000E 03	-0.
W3	0.29078E 04	-0.	0.90000E 05	0.10000E 02	0.10000E 03	-0.
R1	0.39800E 05	-0.	0.90000E 05	0.30000E 02	0.30000E 03	-0.
R2	0.42400E 05	-0.	0.90000E 05	0.30000E 02	0.30000E 03	-0.
R3	0.46000E 05	-0.	0.90000E 05	0.30000E 02	0.30000E 03	-0.
MS	0.53100E 03	-0.	0.90000E 05	0.50000E 01	0.10000E 02	-0.
CSF3C	0.64000E-01	-0.	0.10000E 01	0.50000E-03	0.50000E-02	-0.
TS	0.27440E 03	0.27439E 03	0.27941E 03	0.50000E-03	0.50000E-02	-0.
TSF1I	0.25000E 03	0.24999E 03	0.25501E 03	0.50000E-03	0.50000E-02	-0.
TLF3I	0.85000E 02	0.84990E 02	0.85010E 02	0.50000E-03	0.50000E-02	-0.

***** DEPENDENT VARIABLES *****

VARIABLE NAME	INPUT	MIN. LIMIT	MAX. LIMIT	DELTA	ERROR LIMIT
TOTAL COST	0.28671E-00	-0.	-0.	0.10000E 01	1.00000E-03
PRODUCTION	0.83400E 04	0.83400E 04	0.83400E 04	0.10000E 01	1.00000E-03
FEED	0.18406E 05	-0.	-0.	0.10000E 01	1.00000E-03
TSF1C	0.20426E 03	-0.	0.27940E 03	0.10000E 01	0.50000E 00
TSF2C	0.15442E 03	-0.	-0.	0.10000E 01	0.50000E 00
TSF3C	0.10385E 03	0.85001E 02	-0.	0.10000E 01	0.50000E 00
TC1C	0.20192E 03	-0.	-0.	0.10000E 01	0.50000E 00
TC2C	0.15201E 03	-0.	-0.	0.10000E 01	0.50000E 00
TC3C	0.10155E 03	-0.	-0.	0.10000E 01	0.50000E 00
Q	0.49315E 06	-0.	-0.	0.10000E 01	0.10000E 03
AREA 0	0.33767E 02	-0.	-0.	0.10000E 01	0.10000E-00
AREA 1	0.99277E 03	-0.	-0.	0.10000E 01	0.10000E 01
AREA 2	0.11748E 04	-0.	-0.	0.10000E 01	0.10000E 01
AREA 3	0.12932E 04	-0.	-0.	0.10000E 01	0.10000E 01
COOL W	0.87688E 04	-0.	-0.	0.10000E 01	0.10000E 01
F COST	0.32436E-01	-0.	-0.	0.10000E 01	1.00000E-03
MS COST	0.13275E-00	-0.	-0.	0.10000E 01	1.00000E-03
R1 COST	0.65645E-02	-0.	-0.	0.10000E 01	1.00000E-03
R2 COST	0.33268E-02	-0.	-0.	0.10000E 01	1.00000E-03
R3 COST	0.13086E-02	-0.	-0.	0.10000E 01	1.00000E-04
CW COST	0.52473E-02	-0.	-0.	0.10000E 01	1.00000E-03
A0 COST	0.12696E-02	-0.	-0.	0.10000E 01	1.00000E-04

A1 COST	0.23797E-01	-0.	-0.	0.10000E 01	1.00000E-03
A2 COST	0.28161E-01	-0.	-0.	0.10000E 01	1.00000E-03
A3 COST	0.30999E-01	-0.	-0.	0.10000E 01	1.00000E-03

RUN NUMBER 1 PROBLEM NUMBER 3 LOOP NUMBER 9 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 1	LOOP 2	LOOP 3	LOOP 4	LOOP 5	LOOP 6	LOOP 7	LOOP 8	LOOP 9
INDEPENDENT VARIABLES										
W1	2602.320	2602.320M	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320
W2	2829.890	2829.830	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860
W3	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790
R1	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000
R2	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000
R3	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000
MS	531.000	511.000M	531.000M	511.000M	531.000M	511.000M	521.000M	521.000	521.000	521.000
CSF3C	0.064	0.068	0.064	0.068	0.063	0.068	0.063M	0.067M	0.062M	0.066M
TS	274.400	274.390M	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L
TSF1I	250.000	250.010M	250.020M	250.030M	250.040M	250.050M	250.055M	250.060M	250.065M	250.070M
TLF31	85.000	84.990M	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES										
TOTAL COST	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287
PRODUCTION	8340.000	8339.990U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U
FEED	18405.517	17066.624	18539.243	17180.424	18689.265	17304.431	18845.933	17450.242	19043.581	17622.166
TSF1C	204.261	203.168	204.388	203.284	204.528	203.408	204.667	203.539	204.833	203.692
TSF2C	154.418	152.122	154.663	152.342	154.933	152.580	155.209	152.844	155.545	153.152
TSF3C	103.854	100.352	104.216	100.677	104.618	101.029	105.029	101.426	105.536	101.890
TC1C	201.922	200.813	202.051	200.930	202.192	201.056	202.333	201.189	202.501	201.344
TC2C	152.014	149.671	152.263	149.896	152.538	150.138	152.818	150.408	153.160	150.723
TC3C	101.552	97.914	101.926	98.252	102.340	98.617	102.765	99.031	103.287	99.513
Q	0.49E 06	0.47E 06	0.49E 06	0.47E 06	0.49E 06	0.47E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06
AREA 0	33.767	32.590	33.814	32.623	33.850	32.656	33.336	33.224	33.364	33.251
AREA 1	992.768	1026.255	995.774	1028.955	999.171	1031.906	1034.936	1002.390	1039.622	1006.746
AREA 2	1174.829	1226.623	1177.642	1229.019	1180.856	1231.662	1223.841	1193.582	1228.297	1197.187
AREA 3	1293.249	1383.471	1293.907	1383.038	1294.751	1382.678	1339.315	1335.016	1340.621	1334.998
COOL W	8768.806	15149.891	8103.929	14360.714	7403.543	13539.056	6242.546	13204.610	5419.582	12181.788
F COST	0.032	0.030	0.033	0.030	0.033	0.031	0.033	0.031	0.034	0.031
MS COST	0.133	0.128	0.133	0.128	0.133	0.128	0.130	0.130	0.130	0.130
R1 COST	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
CW COST	0.005	0.009	0.005	0.009	0.004	0.008	0.004	0.008	0.003	0.007
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.024	0.025	0.024	0.025	0.024	0.025	0.025	0.024	0.025	0.024
A2 COST	0.023	0.029	0.028	0.029	0.028	0.030	0.029	0.029	0.029	0.029
A3 COST	0.031	0.033	0.031	0.033	0.031	0.033	0.032	0.032	0.032	0.032

MODIF. S.P. * 100.	0.	-0.004	-0.003	-0.004	-0.003	-0.004	-0.003	-0.003	-0.003
MOVE LIMIT FACTOR	1.000	1.000	1.000	1.000	1.000	0.500	0.500	0.500	0.500

RUN NUMBER 1 PROBLEM NUMBER 3 LOOP NUMBER 18 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 10	LOOP 11	LOOP 12	LOOP 13	LOOP 14	LOOP 15	LOOP 16	LOOP 17	LOOP 18
INDEPENDENT VARIABLES										
W1	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320
W2	2829.890	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860
W3	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790
R1	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000
R2	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000
R3	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000
MS	531.000	521.000	521.000	521.000	521.000	521.000	521.000	521.000	521.000	521.000
CSF3C	0.064	0.062	0.064M	0.067M	0.064M	0.067M	0.064M	0.065M	0.064M	0.065M
TS	274.400	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L
TSF1I	250.000	250.075M	250.077M	250.080M	250.082M	250.085M	250.087M	250.089M	250.090M	250.091M
TLF3I	85.000	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES										
TOTAL COST	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287
PRODUCTION	8340.000	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U
FEED	13405.517	19256.385	18323.021	17536.693	18323.022	17536.694	18323.022	17913.739	18323.022	17913.740
TSF1C	204.261	205.010	204.272	203.631	204.277	203.636	204.282	203.950	204.284	203.953
TSF2C	154.418	155.904	154.357	153.014	154.362	153.019	154.367	153.673	154.369	153.675
TSF3C	103.854	106.077	103.720	101.675	103.725	101.680	103.730	102.673	103.733	102.675
TC1C	201.922	202.680	201.932	201.282	201.937	201.287	201.942	201.606	201.945	201.609
TC2C	152.014	153.525	151.951	150.582	151.956	150.587	151.961	151.253	151.963	151.256
TC3C	101.552	103.845	101.411	99.289	101.416	99.294	101.421	100.325	101.424	100.328
Q	0.49E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06
AREA 0	33.767	33.393	33.319	33.255	33.325	33.261	33.331	33.297	33.333	33.300
AREA 1	992.768	1044.690	1022.628	1004.378	1022.628	1004.378	1022.628	1013.090	1022.628	1013.090
AREA 2	1174.829	1233.137	1212.248	1195.390	1212.249	1195.391	1212.248	1203.383	1212.249	1203.384
AREA 3	1293.249	1342.187	1336.610	1334.985	1336.610	1334.985	1336.610	1335.358	1336.610	1335.357
COOL W	8763.806	4572.540	8538.171	12656.578	8530.985	12647.536	8523.829	10557.273	8520.272	10553.250
F COST	0.032	0.034	0.032	0.031	0.032	0.031	0.032	0.032	0.032	0.032
MS COST	0.133	0.130	0.130	0.130	0.130	0.130	0.130	0.130	0.130	0.130
R1 COST	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
CW COST	0.005	0.003	0.005	0.008	0.005	0.008	0.005	0.006	0.005	0.006
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.024	0.025	0.025	0.024	0.025	0.024	0.025	0.024	0.025	0.024
A2 COST	0.028	0.030	0.029	0.029	0.029	0.029	0.029	0.029	0.029	0.029
A3 COST	0.031	0.032	0.032	0.032	0.032	0.032	0.032	0.032	0.032	0.032

MCDIF. S.P. * 100.	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003
MOVE LIMIT FACTOR	0.500	0.250	0.250	0.250	0.250	0.250	0.125	0.125	0.125

RUN NUMBER 1 PROBLEM NUMBER 3 LOOP NUMBER 25 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 19	LOOP 20	LOOP 21	LOOP 22	LOOP 23	LOOP 24	LOOP 25
INDEPENDENT VARIABLES								
W1	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320	2602.320
W2	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860	2829.860
W3	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790	2907.790
R1	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000	39800.000
R2	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000	42400.000
R3	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000	46000.000
MS	531.000	521.000	521.000	521.000	521.000	521.000	521.000	521.000
CSF3C	0.064	0.064M	0.065M	0.065M	0.064M	0.065M	0.064M	0.065M
TS	274.400	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L	274.390L
TSF1I	250.000	250.092M	250.094M	250.094M	250.095M	250.096M	250.096M	250.097M
TLF3I	85.000	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES								
TOTAL COST	0.287	0.287	0.287	0.287	0.287	0.287	0.287	0.287
PRODUCTION	8340.000	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U
FEED	18405.517	18323.023	17913.740	18114.099	18323.023	18114.099	18323.023	18114.099
TSF1C	204.261	204.287	203.955	204.119	204.289	204.121	204.290	204.122
TSF2C	154.418	154.372	153.678	154.020	154.374	154.021	154.376	154.023
TSF3C	103.854	103.735	102.678	103.199	103.738	103.200	103.739	103.201
TC1C	201.922	201.947	201.611	201.778	201.950	201.779	201.951	201.780
TC2C	152.014	151.966	151.258	151.607	151.968	151.608	151.969	151.610
TC3C	101.552	101.426	100.330	100.870	101.429	100.872	101.430	100.873
Q	0.49E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06	0.48E 06
AREA 0	33.767	33.336	33.303	33.321	33.339	33.322	33.341	33.324
AREA 1	992.768	1022.628	1013.090	1017.749	1022.628	1017.749	1022.628	1017.749
AREA 2	1174.829	1212.249	1203.383	1207.699	1212.249	1207.699	1212.248	1207.699
AREA 3	1293.249	1336.610	1335.358	1335.868	1336.610	1335.868	1336.611	1335.868
COOL W	8768.806	8516.671	10549.244	9525.692	8513.074	9523.771	8511.333	9521.878
F COST	0.032	0.032	0.032	0.032	0.032	0.032	0.032	0.032
MS COST	0.133	0.130	0.130	0.130	0.130	0.130	0.130	0.130
R1 COST	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.007
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
CW COST	0.005	0.005	0.006	0.006	0.005	0.006	0.005	0.006
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.024	0.025	0.024	0.024	0.025	0.024	0.025	0.024
A2 COST	0.028	0.029	0.029	0.029	0.029	0.029	0.029	0.029
A3 COST	0.031	0.032	0.032	0.032	0.032	0.032	0.032	0.032

MODIF. S.P. * 100.	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003	-0.003
MOVE LIMIT FACTOR	0.125	0.125	0.062	0.062	0.062	0.062	0.062

OPTIMUM REACHED ON LOOP 25

RUN NUMBER 1 PROBLEM NUMBER 3 LOOP NUMBER 25 12/12/66

	INPUT VALUE	FINAL VALUE	FINAL-INPUT	MODIF. S.P.	MIN. LIMIT	MAX. LIMIT	MOVE LIMIT
X 1 W1	2602.3200	2602.3200	0.	0.0000	-0.	90000.0000	12.50000000
X 2 W2	2829.8900	2829.8600	-0.0299	0.	-0.	90000.0000	12.50000000
X 3 W3	2907.7900	2907.7900	0.	0.0000	-0.	90000.0000	12.50000000
X 4 R1	39800.0000	39800.0000	0.	0.	-0.	90000.0000	37.50000000
X 5 R2	42400.0000	42400.0000	0.	0.	-0.	90000.0000	37.50000000
X 6 R3	46000.0000	46000.0000	0.	0.	-0.	90000.0000	37.50000000
X 7 MS	531.0000	521.0000	-10.0000	0.0000	-0.	90000.0000	1.25000000
X 8 CSF3C	0.0640	0.0649	0.0009	0.0358	-0.	1.0000	0.00062500
X 9 TS	274.4000	274.3900	-0.0100	-0.0001	274.3900	279.4100	0.00062500
X 10 TSF11	250.0000	250.0968	0.0968	0.0006	249.9900	255.0100	0.00062500
X 11 TLF3I	85.0000	84.9900	-0.0100	-0.0009	84.9900	85.0100	0.00062500
Y 1 TOTAL COST	0.2867	0.2866	-0.0001				
Y 2 PRODUCTION	8339.9999	8339.9700	-0.0299	-0.0000	8339.9700	8339.9900	
Y 3 ELEC	18405.5171	18114.0994	-291.4177	0.	-0.	-0.	
Y 4 TSF1C	204.2605	204.1219	-0.1387	0.	-0.	279.4000	
Y 5 TSF2C	154.4179	154.0226	-0.3953	0.	-0.	-0.	
Y 6 TSF3C	103.8539	103.2011	-0.6529	0.	85.0010	-0.	
Y 7 TC1C	201.9221	201.7801	-0.1420	0.	-0.	-0.	
Y 8 TC2C	152.0142	151.6096	-0.4046	0.	-0.	-0.	
Y 9 TC3C	101.5522	100.8728	-0.6794	0.	-0.	-0.	
Y 10 C	0.493E 06	0.484E 06	-9283.4531	0.	-0.	-0.	
Y 11 AREA 0	33.7668	33.3239	-0.4429	0.	-0.	-0.	
Y 12 AREA 1	992.7676	1017.7491	24.9815	0.	-0.	-0.	
Y 13 AREA 2	1174.8289	1207.6991	32.8701	0.	-0.	-0.	
Y 14 AREA 3	1293.2486	1335.8677	42.6190	0.	-0.	-0.	
Y 15 COLL W	8768.8059	9521.8782	753.0723	0.	-0.	-0.	
Y 16 F COST	0.0325	0.0320	-0.0005	0.	-0.	-0.	
Y 17 MS COST	0.1327	0.1302	-0.0025	0.	-0.	-0.	
Y 18 R1 COST	0.0066	0.0066	0.0000	0.	-0.	-0.	
Y 19 R2 COST	0.0033	0.0033	0.0000	0.	-0.	-0.	
Y 20 R3 COST	0.0013	0.0013	-0.0000	0.	-0.	-0.	
Y 21 CW COST	0.0052	0.0057	0.0005	0.	-0.	-0.	
Y 22 A0 COST	0.0013	0.0013	-0.0000	0.	-0.	-0.	
Y 23 A1 COST	0.0238	0.0244	0.0006	0.	-0.	-0.	
Y 24 A2 COST	0.0282	0.0289	0.0008	0.	-0.	-0.	
Y 25 A3 COST	0.0310	0.0320	0.0010	0.	-0.	-0.	

CYCX MATRIX X 1 X 2 X 3 X 4 X 5 X 6 X 7 X 8 X 9

Y 1	-0.3992E-04	-0.3147E-04	-0.3133E-04	-0.9444E-07	-0.4098E-07	-0.1919E-07	0.2415E-05	0.3581E-01	-0.6333E-04
Y 2	0.1000E 01	0.1000E 01	0.1000E 01	0.	0.	0.	0.	0.	0.
Y 3	0.2197E 01	0.2197E 01	0.2197E 01	0.	0.	0.	0.	-0.3415E 06	0.
Y 4	-0.1624E-01	0.1771E-02	0.1771E-02	0.8063E-03	0.	0.	0.	-0.2754E 03	0.
Y 5	-0.1519E-01	-0.1438E-01	0.3705E-02	0.8063E-03	0.8804E-03	0.	0.	-0.5761E 03	0.
Y 6	-0.1413E-01	-0.1332E-01	-0.1222E-01	0.8063E-03	0.8804E-03	0.8820E-03	0.	-0.8773E 03	0.
Y 7	-0.1629E-01	0.1797E-02	0.1797E-02	0.8058E-03	0.	0.	0.	-0.2793E 03	0.
Y 8	-0.1522E-01	-0.1441E-01	0.3775E-02	0.8063E-03	0.8798E-03	0.	0.	-0.5869E 03	0.
Y 9	-0.1413E-01	-0.1333E-01	-0.1220E-01	0.8063E-03	0.8804E-03	0.8811E-03	0.	-0.9080E 03	0.
Y 10	0.	0.	0.	0.	0.	0.	0.9287E 03	0.	-0.3711E 03
Y 11	0.1843E-03	0.1843E-03	0.1843E-03	0.8392E-04	0.	0.	0.5463E-01	-0.2865E 02	-0.1196E 01
Y 12	0.5396E 00	0.5140E-01	0.5140E-01	0.2588E-01	0.	0.	-0.3276E 01	-0.7995E 04	0.1572E 01
Y 13	0.7070E-01	0.6060E 00	0.4806E-01	-0.1704E-04	0.3028E-01	0.	-0.4052E 01	-0.7476E 04	0.1526E 01
Y 14	0.6275E-01	0.6308E-01	0.6528E 00	-0.1017E-04	-0.2009E-04	0.3162E-01	-0.4525E 01	-0.1381E 04	0.1801E 01
Y 15	0.1815E 02	0.1700E 02	0.1541E 02	-0.1154E 01	-0.1260E 01	-0.1262E 01	0.4953E 02	0.1611E 07	-0.1990E 02
Y 16	0.3878E-05	0.3878E-05	0.3878E-05	0.	0.	0.	0.	-0.6028E 00	0.
Y 17	0.	0.	0.	0.	0.	0.	0.2500E-03	0.	0.
Y 18	0.1557E-05	-0.1698E-06	-0.1698E-06	0.3813E-07	0.	0.	0.	0.2640E-01	0.
Y 19	-0.1060E-05	0.7478E-06	0.3476E-07	0.5054E-07	0.4386E-07	0.	0.	-0.5405E-02	0.
Y 20	-0.4586E-06	-0.4351E-06	0.3236E-06	0.2360E-07	0.2577E-07	0.1652E-07	0.	-0.1279E-01	0.
Y 21	0.1086E-04	0.1017E-04	0.9221E-05	-0.6906E-06	-0.7541E-06	-0.7552E-06	0.2964E-04	0.9638E 00	-0.1193E-04
Y 22	0.6929E-08	0.6929E-08	0.6929E-08	0.3155E-08	0.	0.	0.2054E-05	-0.1077E-02	-0.4498E-04
Y 23	0.1293E-04	0.1232E-05	0.1232E-05	0.6203E-06	0.	0.	-0.7852E-04	-0.1916E-00	0.3749E-04
Y 24	0.1635E-05	0.1453E-04	0.1152E-05	-0.4113E-09	0.7259E-06	0.	-0.9712E-04	-0.1792E-00	0.3679E-04
Y 25	0.1504E-05	0.1512E-05	0.1565E-04	-0.2406E-09	-0.4812E-09	0.7578E-06	-0.1085E-03	-0.3311E-01	0.4331E-04

CYDX MATRIX	X 10	X 11	X
Y 1	0.6109E-03	-0.8568E-03	
Y 2	0.	0.	
Y 3	0.	0.	
Y 4	0.1001E 01	0.	
Y 5	0.1001E 01	0.	
Y 6	0.1001E 01	0.	
Y 7	0.1001E 01	0.	
Y 8	0.1001E 01	0.	
Y 9	0.1001E 01	0.	
Y 10	0.	0.	
Y 11	0.1173E 01	0.	
Y 12	0.	0.	
Y 13	-0.5188E 00	0.	

```
Y 14        0.6104E 00   0.
Y 15       -0.1433E 04   0.1432E 04
Y 16        0.           0.
Y 17        0.           0.
Y 18        0.1064E-03   0.
Y 19        0.6279E-04   0.
Y 20        0.2954E-04   0.
Y 21       -0.8574E-03   0.8568E-03
Y 22        0.4411E-04   0.
Y 23        0.           0.
Y 24       -0.1281E-04   0.
Y 25        0.1444E-04   0.

SECONDS =     0
```

RUN NUMBER 2 PROBLEM NUMBER 1 LOOP NUMBER 9 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 1	LOOP 2	LOOP 3	LOOP 4	LOOP 5	LOOP 6	LOOP 7	LOOP 8	LOOP 9
INDEPENDENT VARIABLES										
W1	2231.164	2231.164M	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164
W2	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276
W3	3682.559	3682.550	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530
R1	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520
R2	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700
R3	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250
MS	455.266	475.266M	495.266M	515.266M	495.266M	515.266M	505.266M	505.266	515.266M	505.266M
CSF3C	0.054	0.056	0.054	0.056	0.054	0.056	0.055	0.053M	0.055M	0.053M
TS	274.400	274.390M	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U
TSF1I	250.000	250.010M	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U
TLF31	85.000	84.990M	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES										
TOTAL COST	0.293	0.290	0.289	0.290	0.289	0.290	0.239	0.290	0.289	0.290
PRODUCTION	8339.999	8339.990U	8339.970U	8339.970U	8339.970U	8339.970U	3339.970U	8339.970U	3339.970U	8339.970U
FEED	23703.155	22358.057	23601.991	22259.045	23456.010	22144.667	23302.428	24844.603	23264.213	24794.444
TSF10	210.652	209.706	210.592	209.634	210.490	209.550	210.332	211.439	210.355	211.405
TSF2C	167.787	165.798	167.650	165.647	167.437	165.471	167.212	169.421	167.155	169.351
TSF3C	102.752	99.179	102.499	98.909	102.117	98.595	101.713	105.672	101.612	105.546
TC1C	208.379	207.426	208.318	207.353	208.216	207.268	208.107	209.171	208.080	209.138
TC2C	165.551	163.543	165.413	163.391	165.198	163.214	164.970	167.198	164.913	167.127
TC3C	100.771	97.144	100.515	96.869	100.127	96.550	99.717	103.731	99.615	103.604
Q	0.42E 06	0.44E 06	0.46E 06	0.48E 06	0.46E 06	0.48E 06	0.47E 06	0.47E 06	0.48E 06	0.47E 06
AREA 0	29.550	30.596	31.797	32.781	31.786	32.771	32.323	32.445	32.966	32.441
AREA 1	1045.793	939.174	904.333	821.008	901.125	818.741	868.920	901.717	841.107	900.638
AREA 2	1202.444	1082.216	1037.548	944.017	1034.226	941.678	997.110	1031.132	964.205	1030.010
AREA 3	2122.097	1918.758	1806.678	1652.325	1803.034	1649.881	1735.236	1773.732	1674.746	1772.430
COOL W	1045.224	9945.829	3613.591	13343.734	4369.166	14292.626	5753.395	-1372.854	6527.323	-1179.791
F COST	0.042	0.039	0.042	0.039	0.041	0.039	0.041	0.044	0.041	0.044
MS COST	0.114	0.119	0.124	0.129	0.124	0.129	0.126	0.126	0.129	0.126
R1 COST	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002
CW COST	0.001	0.006	0.002	0.008	0.003	0.009	0.003	-0.001	0.004	0.001
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.025	0.023	0.022	0.020	0.022	0.020	0.021	0.022	0.020	0.022
A2 COST	0.029	0.026	0.025	0.023	0.025	0.023	0.024	0.025	0.023	0.025
A3 COST	0.051	0.046	0.043	0.040	0.043	0.040	0.042	0.043	0.040	0.042

RUN NUMBER 2 PROBLEM NUMBER 1 LOOP NUMBER 18 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 10	LOOP 11	LOOP 12	LOOP 13	LOOP 14	LOOP 15	LOOP 16	LOOP 17	LOOP 18
INDEPENDENT VARIABLES										
W1	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164
W2	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276
W3	3682.559	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530
R1	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520
R2	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700
R3	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250
MS	455.266	515.266M	510.266M	515.266M	510.266M	515.266M	510.266M	512.766M	512.766	510.266M
CSF3C	0.054	0.055	0.053	0.055	0.053	0.055	0.053	0.054M	0.055M	0.054M
TS	274.400	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U
TSF1I	250.000	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U
TLF3I	85.000	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES										
TOTAL COST	0.293	0.289	0.290	0.289	0.290	0.289	0.290	0.289	0.289	0.289
PRODUCTION	8339.999	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U
FEED	23703.155	23222.669	24739.554	23174.004	24675.825	23121.176	24605.854	23546.628	22616.920	23546.628
TSF1C	210.652	210.326	211.369	210.292	211.326	210.254	211.279	210.553	209.894	210.553
TSF2C	167.787	167.094	169.275	167.022	169.185	166.944	169.097	167.570	166.190	167.570
TSF3C	102.752	101.502	105.409	101.373	105.249	101.233	105.073	102.354	99.882	102.354
TC1O	203.379	208.050	209.101	208.016	209.058	207.978	209.010	208.279	207.615	208.279
TC2O	165.551	164.852	167.050	164.779	166.960	164.700	166.861	165.331	163.939	165.331
TC3O	100.771	99.503	103.464	99.373	103.302	99.230	103.124	100.368	97.859	100.368
Q	0.42E 06	0.48E 06	0.47E 06	0.48E 06	0.47E 06	0.48E 06	0.47E 06	0.48E 06	0.48E 06	0.47E 06
AREA 0	29.550	32.863	32.712	32.859	32.707	32.854	32.702	32.753	32.676	32.617
AREA 1	1045.793	840.270	885.101	839.290	883.760	838.227	882.239	853.464	834.592	860.220
AREA 2	1202.444	963.937	1012.147	962.923	1010.752	961.822	1009.224	978.457	958.921	986.320
AREA 3	2122.097	1673.738	1740.128	1672.670	1738.510	1671.457	1736.737	1695.883	1674.523	1710.694
COOL W	1045.224	6767.776	-736.742	7053.236	-482.605	7367.931	-199.550	4832.651	10493.423	4698.937
F COST	0.042	0.041	0.044	0.041	0.044	0.041	0.043	0.042	0.040	0.042
MS COST	0.114	0.129	0.128	0.129	0.128	0.129	0.128	0.128	0.128	0.128
R1 COST	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002
CW COST	0.001	0.004	-0.000	0.004	-0.000	0.004	-0.000	0.003	0.006	0.003
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.025	0.020	0.021	0.020	0.021	0.020	0.021	0.020	0.020	0.021
A2 COST	0.029	0.023	0.024	0.023	0.024	0.023	0.024	0.023	0.023	0.024
A3 COST	0.051	0.040	0.042	0.040	0.042	0.040	0.042	0.041	0.040	0.041

RUN NUMBER 2 PROBLEM NUMBER 1 LOOP NUMBER 25 12/12/66

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 19	LOOP 20	LOOP 21	LOOP 22	LOOP 23	LOOP 24	LOOP 25
INDEPENDENT VARIABLES								
W1	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164	2231.164
W2	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276	2426.276
W3	3682.559	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530	3682.530
R1	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520	34123.520
R2	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700	36352.700
R3	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250	39439.250
MS	455.266	510.266	510.266	510.266	510.266	510.266	510.266	510.266
CSF3C	0.054	0.055M	0.054M	0.055M	0.054M	0.055M	0.054M	0.055M
TS	274.400	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U	274.390U
TSF1I	250.000	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U	250.010U
TLF3I	85.000	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U	84.990U
DEPENDENT VARIABLES								
TOTAL COST	0.293	0.289	0.289	0.289	0.289	0.289	0.289	0.289
PRODUCTION	8339.999	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U	8339.970U
FEED	23703.155	22616.920	23546.628	23067.116	23546.629	23067.116	23546.629	23067.117
TSF1C	210.652	209.894	210.553	210.216	210.553	210.216	210.553	210.216
TSF2C	167.787	166.190	167.570	166.864	167.570	166.864	167.570	166.864
TSF3C	102.752	99.882	102.354	101.090	102.354	101.090	102.354	101.090
TC1C	208.379	207.615	208.279	207.939	208.279	207.939	208.279	207.939
TC2C	165.551	163.939	165.331	164.619	165.331	164.619	165.331	164.619
TC3C	100.771	97.859	100.368	99.085	100.368	99.085	100.368	99.085
Q	0.42E 06	0.47E 06	0.47E 06	0.47E 06	0.47E 06	0.47E 06	0.47E 06	0.47E 06
AREA 0	29.550	32.540	32.617	32.577	32.617	32.577	32.617	32.577
AREA 1	1045.793	841.158	860.220	850.353	860.220	850.353	860.220	850.353
AREA 2	1202.444	966.593	986.320	976.100	986.320	976.100	986.320	976.101
AREA 3	2122.097	1689.180	1710.695	1699.427	1710.694	1699.427	1710.695	1699.427
COOL W	1045.224	10337.552	4698.937	7406.746	4698.912	7406.723	4698.912	7406.698
F COST	0.042	0.040	0.042	0.041	0.042	0.041	0.042	0.041
MS COST	0.114	0.128	0.128	0.128	0.128	0.128	0.128	0.128
R1 COST	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005
R2 COST	0.003	0.003	0.003	0.003	0.003	0.003	0.003	0.003
R3 COST	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002
CW COST	0.001	0.006	0.003	0.004	0.003	0.004	0.003	0.004
A0 COST	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
A1 COST	0.025	0.020	0.021	0.020	0.021	0.020	0.021	0.020
A2 COST	0.029	0.023	0.024	0.023	0.024	0.023	0.024	0.023
A3 COST	0.051	0.040	0.041	0.041	0.041	0.041	0.041	0.041

[illegible] 3 [illegible] 1 LOOP NUMBER 9 ?/23/67

SUMMARY OF INDIVIDUAL LOOPS

[illegible] VARIABLES	INPUT	LOOP 1	LOOP 2	LOOP 3	LOOP 4	LOOP 5	LOOP 6	LOOP 7	LOOP 8	[illegible]
[illegible]1	3??3.?00	2?65.110M	2??8.420M	2?52.?70M	2420.105M	?173.915M	2069.999	26?7.356M	?706.?11M	[illegible]
[illegible]	4?14.?10	1??7.7?3M	1?70.2721	1?70.272	2207.?96M	2217.?26	2270.035M	234?.471M	23?6.??1M	[illegible]
[illegible]	1?1.700	4146.?66M	4070.?51M	3816.452M	3641.550M	4058.744M	4001.500M	3302.?34M	324?.?6?	[illegible]
[illegible]1	5?7??.?1??	?423.?00M	?369.8??M	?6552.42?M	?5224.740M	?4332.33?M	?54259.264M	?53323.?33M	?53245.44?	[illegible]
[illegible]	?1690.?0??	?699.919M	?1282.952M	?8965.?94	?33728.130M	?29044.929M	?35727.372M	?34739.379M	?34795.971M	[illegible]
[illegible]	69999.?99?	?632.952M	?655.920M	?6472.975M	?6?134.098M	?2858.193M	?2914.294M	?62941.944M	?61995.405M	[illegible]
[illegible]	755.500	465.4934	470.465	479.857	503.016	562.???	570.499	706.6?6	7?0.0??	[illegible]
CS[illegible]	0.096	0.1?2M	0.120M	0.117M	0.107M	0.082M	0.079M	0.052M	0.05?M	[illegible]
TS	274.400	274.400U	274.400U	274.400U	274.400U	274.400U	274.400U	274.400U	274.400U	274.400U
TSE[illegible]	250.000	250.000U	250.000U	250.000U	250.010U	250.010U	250.010U	250.000U	250.000U	[illegible]
TI[illegible]	35.000	35.100U	85.000U	35.000U	85.000U	85.000U	35.000U	85.000U	85.010U	[illegible]

[illegible] VARIABLES	INPUT	LOOP 1	LOOP 2	LOOP 3	LOOP 4	LOOP 5	LOOP 6	LOOP 7	LOOP 8	[illegible]
TOTAL COST	0.334	0.987	0.719	0.400	0.357	0.310	0.306	0.313	0.316	[illegible]
O[illegible] FUNCTION	8340.000	8330.718U	8339.481U	8339.793U	8339.749U	8339.754U	8339.684U	8339.711U	8342.093U	[illegible]
F[illegible]	13125.246	11600.371	11754.392	11913.439	12373.954	14534.7??	14995.537	20975.236	20779.9??	?9?72.1??
TS[illegible]10	194.?99	214.139	215.699	211.996	213.034	219.446	219.698	213.150	212.7?1	[illegible]
TS[illegible]20	134.951	179.147	176.076	179.367	166.549	174.7??	171.318	157.242	166.?94	[illegible]
TS[illegible]30	132.256	121.621	119.607	117.311	115.954	117.639	115.915	124.564	123.75?	122.??
TC10	1?2.2?3	213.747	213.244	209.528	210.590	217.096	217.354	210.336	210.434	[illegible]
TC20	131.?41	176.595	173.516	167.766	163.953	172.3??	169.524	165.537	164.131	[illegible]
TC30	12?.3??	117.533	115.570	113.379	112.310	114.776	113.160	122.438	121.41?	[illegible]
[illegible]	0.74E 06	0.42E 06	0.44E 06	0.45E 06	0.47E 06	0.52E 06	0.53E 06	0.65E 06	0.6?E 06	[illegible]
[illegible]	49.293	30.742	31.133	31.593	32.355	36.322	36.316	44.553	45.?77	[illegible]
AREA 1	1?03.420	1?07.?34	1?94.555	1556.529	1370.754	?09.371	?72.915	?23.?21	?8?.?1	[illegible]
AREA 2	1705.9?6	?29.534	784.911	713.362	765.775	651.73?	?26.301	535.537	51?.?1?	[illegible]
AREA 3	?63.273	?1?69.525	?0444.207	10721.267	5443.153	2??0.135	?692.974	1312.157	121?.??	[illegible]
[illegible]	31?1.?1?	1?4?.52?	1?41.460	2??3.7?4	3690.726	21?2.543	2893.139	-30?0.???	-2816.???	[illegible]
[illegible] COST	0.0?3	0.021	0.021	0.021	0.02?	0.0??	0.026	0.037	0.037	[illegible]
[illegible]	0.1??	0.116	0.118	0.120	0.126	0.141	0.143	0.177	0.1??	[illegible]
[illegible] COST	0.011	0.0??	0.010	0.008	0.00?	0.007	0.007	0.00?	0.0??	[illegible]
[illegible] COST	0.00?	0.004	0.003	0.003	0.003	0.00?	0.004	0.013	0.0?	[illegible]
[illegible] COST	0.00?	0.003	0.003	0.003	0.002	0.003	0.003	0.0??	[illegible]	[illegible]
[illegible] COST	0.0??	0.001	0.001	0.00?	0.00?	0.001	0.00?	-0.0??	[illegible]	[illegible]
A? COST	0.0??	0.001	0.001	0.001	0.001	0.0??	0.001	0.0??	[illegible]	[illegible]
A1 COST	0.0??	0.023	0.033	0.0?7	0.03?	0.0??	0.021	0.0??	0.0??	[illegible]
A2 COST	0.041	0.021	0.019	0.017	0.01?	0.01?	0.015	0.013	0.01?	[illegible]
A3 COST	0.0??	0.759	0.400	0.257	0.130	0.170	0.065	0.131	0.1??	[illegible]

RUN NUMBER 3 PROBLEM NUMBER 1 LOOP NUMBER 11 2/23/67

SUMMARY OF INDIVIDUAL LOOPS

	INPUT	LOOP 10	LOOP 11
INDEPENDENT VARIABLES			
W1	3003.600	2679.833U	2679.833
W2	4233.310	2448.215U	2448.215
W3	101.700	2213.952U	2213.952
A1	59700.000	53111.118U	53111.118
A2	63600.000	34654.539U	34654.539
A3	60000.000	61894.446U	61894.446
WS	796.530	800.518	800.518
CSW3	0.036	0.059U	0.059
TS	274.400	274.400U	274.400U
TSW1	250.000	250.000U	250.000U
TLW2	35.000	85.010U	85.010U
DEPENDENT VARIABLES			
TOTAL COST	0.334	0.326	0.326
PRODUCTION	8340.000	8342.000U	8342.000U
FEED	13125.246	20370.532	20371.189
TSC1	194.919	212.854	212.849
TSC2	134.851	164.962	164.950
TSC3	132.256	122.409	122.392
TC1	192.283	210.525	210.520
TC2	131.261	162.639	162.626
TC3	123.992	120.247	120.230
Q	0.74E 06	0.74E 06	0.74E 06
AREA 0	40.213	49.485	49.484
AREA 1	1203.420	752.812	752.707
AREA 2	1795.965	467.304	467.215
AREA 3	63.272	1041.293	1041.236
COND	2121.315	-13.392*	-0.776*
[illegible] COST	0.123	0.036	0.036
WS COST	0.190	0.200	0.200
W1 COST	0.011	0.008	0.003
W2 COST	0.005	0.003	0.003
W3 COST	0.009	0.002	0.002
CW COST	0.002	-0.001	-0.001
A0 COST	0.002	0.002	0.002
A1 COST	0.020	0.013	0.013
A2 COST	0.041	0.011	0.011
A3 COST	0.012	0.025	0.025

APPENDIX III

DESCRIPTION OF COMPUTER PROGRAM - MODEL

Computer program MODEL may be used for simulation studies of the MEMS process. Furthermore, the program may be used to obtain feasible starting points for SUBROUTINE MODEL which are near the boundary of the feasible constraint region. Table III-1 describes the variables used in MODEL and the SUBROUTINE MODEL used in the optimization study. A logic diagram for program MODEL is given in Figure III-1. Program MODEL is listed in Table III-2.

Table III. Explanation of Computed Program Variables Used In Program MODEL and SUBROUTINE MODEL

Symbol	Explanation
A	Constant used as multiplier to change size of independent variables
AA	The first term of the Lagnangian Polynomial
AB	The second term of the Lagnangian Polynomial
AC	The third term of the Lagnangian Polynomial
AD	The fourth term of the Lagnangian Polynomial
AL	Latent heat of vaporization of distillate water in each effect
ALS	Latent heat of brine heater steam
AL1I	Flashing brine stream flow rate into effect one
AL2I	Flashing brine stream flow rate into effect two
AL3I	Flashing brine stream flow rate into effect three
AL1O	Flashing brine stream flow rate out of effect one
AL2O	Flashing brine stream flow rate out of effect two
AL3O	Flashing brine stream flow rate out of effect three
AN1	Number of stages in effect one

Table 3. (Continued)

Symbol	Explanation
AN2	Number of stages in effect two
AN3	Number of stages in effect three
A1	Average boiling point elevation in effect one
A2	Average boiling point elevation in effect two
A3	Average boiling point elevation in effect three
B	A constant used in the equations which calculate horsepower
CF	Salinity of feed stream
CONST(1) through CONST(7)	Cost coefficients used in the terms of the process objective function
CP	Heat capacity of flashing brine solution
CSF1I	Salinity of flashing brine stream fed to effect one
CSF2I	Salinity of flashing brine stream fed to effect two
CSF3I	Salinity of flashing brine stream fed to effect three
CSF1O	Salinity of flashing brine stream discharged from effect one
CSF1O	Salinity of flashing brine stream discharged from effect two
CSF1O	Salinity of flashing brine stream discharged from effect three

Table 3. (Continued)

Symbol	Explanation
DT0	Temperature difference available for heat transfer in brine
DT1	Temperature difference available for heat transfer in effect one
DT2	Temperature difference available for heat transfer in effect two
DT3	Temperature difference available for heat transfer in effect three
HP1	Horsepower of the recycle pump of effect one
HP2	Horsepower of the recycle pump of effect two
HP3	Horsepower of the recycle pump of effect three
I,J,K	Subscripts and counters used in program MODEL
TLF0I	Temperature indicated by $(T_j)_A$ on Figure 5
TLF1I	Temperature indicated by $(T_j)_C$ on Figure 5
TLF2I	Temperature indicated by $(T_j)_F$ on Figure 5
U0, U1 U2, U3	Overall heat transfer coefficients used in the brine heater and effects one, two and three respectively
XA	230° F
XB	250°F
XC	270°F
XD	290°F
X(1)	Distillate produced in effect one

Table 3. (Continued)

Symbol	Explanation
X(2)	Distillate produced in effect two
X(3)	Distillate produced in effect three
X(4)	Recycle used in effect one
X(5)	Recycle used in effect two
X(6)	Recycle used in effect three
X(7)	Brine heater steam consumption rate
X(8)	Salinity of flashing brine stream discharged from the MEMS plant
X(9)	Brine heater steam temperature
X(10)	Temperature of flashing brine stream fed to effect one
X(11)	Temperature of seawater fed to MEMS plant
YA	Enthalpy of saturated steam at temperature XA
YB	Enthalpy of saturated steam at temperature XB
YC	Enthalpy of saturated steam at temperature XC
YD	Enthalpy of saturated steam at temperature XD
Y(1)	Total cost per 1000 gal. distillate water produced
Y(2)	Total distillate water production rate
Y(3)	Seawater feed rate
Y(4)	Temperature of flashing brine stream leaving effect one

Table 3. (Continued)

Symbol	Explanation
Y(5)	Temperature of flashing brine stream leaving effect two
Y(6)	Temperature of flashing brine stream leaving effect three
Y(7)	Temperature of distillate stream flowing from effect one
Y(8)	Temperature of distillate stream flowing from effect two
Y(9)	Temperature of distillate stream flowing from effect three
Y(10)	Brine heater heat transfer rate
Y(11)	Brine heater heat transfer area
Y(12)	Effect one heat transfer area
Y(13)	Effect two heat transfer area
Y(14)	Effect three heat transfer area
Y(15)	Cooling water flow rate
Y(16)	Cost for feed pretreatment and pumping
Y(17)	Cost of brine heater steam
Y(18)	Cost of recycling brine in effect one
Y(19)	Cost of recycling brine in effect two
Y(20)	Cost of recycling brine in effect three
Y(21)	Cooling water cost
Y(22)	Cost of heat transfer area in brine heater
Y(23)	Cost of heat transfer area in effect one

Table 3. (Continued)

Symbol	Explanation
Y(24)	Cost of heat transfer area in effect two
Y(25)	Cost of heat transfer area in effect three
Z	Dummy variable used in Lagrange polynamial

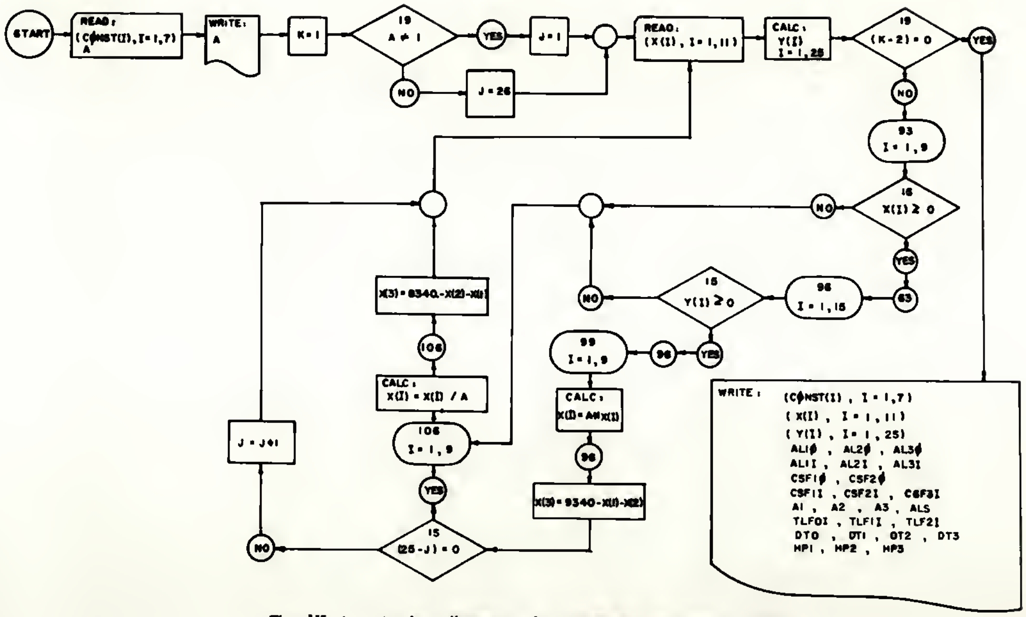

Fig. III-1. Logic diagram for computer program MODEL.

Table III-2. Computer Program MODEL

```
     MONSS      JOB
     MONSS      COMT  8 MINUTES, 9 PAGES,
     MONSS      ASGN MJB,12
     MONSS      ASGN MGO,16
     MONSS      MODE GO
     MONSS      EXEQ FORTRAN,,,,,,,MODEL
      DIMENSIONX(12),Y(25),CONST(7)
C     SIMULATION PROGRAM MEMS SEAWATER DISTILLATION PLANT
    1 FORMAT(20X,F13.6)
    2 FORMAT(20X,E13.6)
    3 FORMAT(5E13.6)
    4 FORMAT(6F10.0)
      READ(1,4)(CONST(I),I=1,7)
    6 CONTINUE
      READ(1,1)A
      WRITE(3,2)A
      K=1
      IF(A.NE.1.)GOTO13
      J=26
      GOTO14
   13 J=1
   14 READ(1,1)(X(I),I=1,11)
   15 CONTINUE
C     SIMULATION PROGRAM MEMS SEAWATER DISTILLATION PLANT
C     ************************************************************
C     CONSTANTS USED IN PERFORMANCE EQUATIONS
      AL=1000.
      B=17.01723
      AN1=23.
      AN2=23.
      AN3=22.
      CF=0.035
      CP=1.
      XA=230.
      XB=250.
      XC=270.
      XD=290.
      YA=958.8
      YB=945.5
      YC=931.8
      YD=917.5
      U0=510.
      U1=510.
      U2=510.
      U3=510.
C     ************************************************************
C     PERFORMANCE EQUATIONS
      Y(2)=X(1)+X(2)+X(3)
```

Table III-2. (Con't)

```
 Y(3)=Y(2)/(1.-CF/X(8))
 AL1O=Y(3)-X(1)
 AL2O=AL1O-X(2)
 AL3O=AL2O-X(3)
 AL1I=Y(3)+X(4)
 AL2I=Y(3)+X(5)-X(1)
 AL3I=Y(3)+X(6)-X(1)-X(2)
 CSF1O=CF*Y(3)/AL1O
 CSF2O=CF*Y(3)/AL2O
 CSF1I=CSF1O*(AL1O+X(4))/AL1I
 CSF2I=CSF2O*(AL2O+X(5))/AL2I
 CSF3I=X(8)*(AL3O+X(6))/AL3I
 Y(4)=X(10)-(AL/CP)*ALOG(CSF1O/CSF1I)
 Y(5)=Y(4)-(AL/CP)*ALOG(CSF2O/CSF2I)
 Y(6)=Y(5)-(AL/CP)*ALOG(X(8)/CSF3I)
 A1=1.0100+(CSF1I+CSF1O)/(2.*0.0300)
 A2=1.0075+(CSF2I+CSF2O)/(2.*0.0347)
 A3=0.3201+(CSF3I+X(8))/(2.*0.0315)
 Y(7)=Y(4)-A1
 Y(8)=Y(5)-A2
 Y(9)=Y(6)-A3
 Z=X(9)
 LAGRANGIAN POLYNOMIAL
 AA=YA*(Z-XB)*(Z-XC)*(Z-XD)/((XA-XB)*(XA-XC)*(XA-XD))
 AB=YB*(Z-XA)*(Z-XC)*(Z-XD)/((XB-XA)*(XB-XC)*(XB-XD))
 AC=YC*(Z-XA)*(Z-XB)*(Z-XD)/((XC-XA)*(XC-XB)*(XC-XD))
 AD=YD*(Z-XA)*(Z-XB)*(Z-XC)/((XD-XA)*(XD-XB)*(XD-XC))
 ALS=AA+AB+AC+AD
 ***********************************************************
 Y(10)=X(7)*ALS
 TLFOI=X(10)-Y(10)/(CP*(Y(3)+X(4)))
 TLF1I=(CP*((AL1O+X(5))*Y(4)+X(1)*Y(7))-Y(10))
1/(CP*(Y(3)+X(5)))
 TLF2I=(CP*((AL2O+X(6))*Y(5)+(X(1)+X(2))*Y(8))-Y(10))
1/(CP*(Y(3)+X(6)))
 Y(15)=(Y(10)+Y(3)*CP*X(11)-(CP*(AL3O*Y(6)+Y(2)*Y(9))))
1/(CP*(Y(6)-X(11)))
 DTO=X(9)-0.5*(X(10)+TLFOI)
 DT1=X(10)-TLFOI-A1-(X(10)-Y(4))/(2.*AN1)
 DT2=Y(4)-TLF1I-A2-(Y(4)-Y(5))/(2.*AN2)
 DT3=Y(5)-TLF2I-A3-(Y(5)-Y(6))/(2.*AN3)
 Y(11)=Y(10)/(DTO*UO)
 Y(12)=X(1)*AL/(DT1*U1)
 Y(13)=X(2)*AL/(DT2*U2)
 Y(14)=X(3)*AL/(DT3*U3)
 HP1=X(4)*B*(EXP(-AL/(0.1104*(X(10)+460.)))
1-EXP(-AL/(0.1104*(Y(4)+460.))))
 HP2=X(5)*B*(EXP(-AL/(0.1104*(Y(4)+460.)))
1-EXP(-AL/(0.1104*(Y(5)+460.))))
```

Table III-2. (Con't)

```
      HP3=X(6)*B*(EXP(-AL/(0.1104*(Y(5)+460.)))
     1-EXP(-AL/(0.1104*(Y(6)+460.))))
C     ****************************************************************
C     COST EQUATIONS
      Y(16)=CONST(1)*Y(3)/1.E+10
      Y(17)=CONST(2)*X(7)/1.E+10
      Y(18)=CONST(3)*HP1/1.E+10
      Y(19)=CONST(3)*HP2/1.E+10
      Y(20)=CONST(3)*HP3/1.E+10
      Y(21)=CONST(4)*Y(15)/1.E+10
      Y(22)=CONST(6)*Y(11)/1.E+10
      Y(23)=CONST(5)*Y(12)/1.E+10
      Y(24)=CONST(5)*Y(13)/1.E+10
      Y(25)=CONST(5)*Y(14)/1.E+10
      Y(1)=Y(16)+Y(17)+Y(18)+Y(19)+Y(20)+Y(21)
     1+Y(22)+Y(23)+Y(24)+Y(25)+CONST(7)/1.E+10
      IF(K-2)91,110,91
   91 DO93I=1,8
      IF(X(I))104,104,94
   93 CONTINUE
   94 DO96I=1,15
      IF(Y(I))104,104,96
   96 CONTINUE
   97 DO99I=1,8
      X(I)=A*X(I)
   99 CONTINUE
      X(3)=8340.-X(2)-X(1)
      IF(25-J)104,102,102
  102 J=J+1
      GOTO15
  104 DO106I=1,8
      X(I)=X(I)/A
  106 CONTINUE
      X(3)=8340.-X(2)-X(1)
      K=2
      GOTO15
  110 WRITE(3,2)(CONST(I),I=1,7)
      WRITE(3,1)(X(I),I=1,11)
      WRITE(3,2)(Y(I),I=1,25)
      WRITE(3,3)AL1O,AL2O,AL3O
      WRITE(3,3)AL1I,AL2I,AL3I
      WRITE(3,3)CSF1O,CSF2O
      WRITE(3,3)CSF1I,CSF2I,CSF3I
      WRITE(3,3)A1,A2,A3,ALS
      WRITE(3,3)TLF0I,TLF1I,TLF2I
      WRITE(3,3)DT0,DT1,DT2,DT3
      WRITE(3,3)HP1,HP2,HP3
      GOTO6
      END
```

Table III-2. (Con't)

```
     MONSS       EXEQ LINKLOAD
                 CALL MODEL
     MONSS       EXEQ MODEL,MJB


      DATA
   17650.0 2500000. 57590000.0     5984.0  239700.0  376000.0
208000000.
                               1.
                            2602.32
                            2829.86
                            2907.79
                           36961.
                           39376.
                           42719.
                             510.
                                .065
                             274.4
                             250.
                              85.
```

PART II.

OPTIMIZATION OF A MULTI-STAGE AERATED LAGOON BY THE DISCRETE MAXIMUM PRINCIPLE

1.0 INTRODUCTION

Pollution abatement has become a subject of increasing concern both to the technical expert in this area and to the common citizen. Sources of pollution range from plant life to large manufacturing complexes. The tools for combatting pollution range from plant life to man made pollution control devices.

In most industries it is not economically feasible to prevent waste formation. As a result these industries must concentrate on destroying or disposing of wastes once they have been formed. The chemical process industry and the petroleum industry serve as examples of this type of industry.

One means of measuring the strength of a pollutant is to determine the total amount of oxygen that is required to reduce the pollutant to a harmless state. Some pollutants react directly with oxygen. The rates of reaction of these pollutants are usually quite rapid. Other pollutants are degraded by bacteria or microorganisms in an oxygen enriched environment. Microorganism feeding processes usually occur at a considerably slower rate than direct oxidation reactions.

This work deals with the modeling and optimization of a system for treating petroleum refinery waste water. The system or process considered is an aerated lagoon. In general the process consists of introducing waste water solutions into a large body of water wherein they are degraded sufficiently to allow the effluent stream to be discharged from the refinery.

The lagoon model used in this paper is partially patterned after the aeration basins of American Oil Company's Sugar Creek Refinery which is located in Sugar Creek Missouri. Several of the constants used in the model

were calculated from operating data from this particular aerated lagoon. A detailed description of the Sugar Creek aerated lagoon, its operating characteristics, and the nature of the wastes it treats is given by Stroud, Sorg, and Lamkin (1) and Burkhead (2). A short description of the Sugar Creek Lagoon is given here. Waste water is first introduced into a pond which has an oil skimmer trough at its outlet for removing any surface oil slick which forms. The effluent from the oil skimming pond is introduced by gravity flow evenly across the inlet of the first aeration basin. The first aeration basin is approximately 713 feet in length by 120 feet in width with a depth of 10 feet.

The effluent from the first basin is introduced by gravity flow into the second aeration basin. This basin is approximately 700 feet in length by 120 feet in width with a depth of 10 feet. The effluent from this basin is introduced by gravity flow into the second aeration basin. This basin is approximately 700 feet in length by 120 feet in width with a depth of 10 feet. The effluent from this basin is introduced into a settling basin before it is discharged from the refinery.

In the first aeration basin, three mechanical surface aerators which are driven by 60 horsepower electric motors are mounted on steel platforms. The platforms are positioned down the basin center line which is parallel to the over all direction of waste water flow. The second aeration basin has three 15 horsepower surface aerators positioned down its center line.

Recently four more 20 horsepower aerators have been added to the first basin. These aerators are currently located along a line which is adjacent to and parallel to the inlet baffle. However, these aerators are not on immobile platforms as are the other aerators. The aerators oxygenate and keep the waste water in the basins mixed.

The optimization goal is to determine the size of lagoon and the sizes and positions of the aerators required to achieve a specified waste water conversion such that the total cost of the aerated lagoon is minimized.

The lagoon is considered to be a stagewise process. The optimum aerator horsepower and the lagoon volume required at each stage are determined by a discrete version of Pontryagin's maximum principle as elucidated by Fan and Wang (3).

2.0 DESCRIPTION OF THE LAGOON MODEL

In this section the kinetic, flow, and economic models of the process are developed.

2.1 Ideal Component Assumption and Kinetic Model.

In the lagoon model a real waste solution which may have many types of impurities is assumed to be composed of wastes which fit into one of three categories. The impurities in each category are further assumed to act as a single idealized component.

The first idealized impurity is a mixture of organic and inorganic impurities which can be degraded to harmless products if it remains in the presence of degrading aerobic bacteria in an oxygenated environment for a sufficient length of time. This component is commonly measured in terms of biological oxygen demand (BOD) which is defined by Eckenfelder and O'Connor (4) as "... that quantity of oxygen required during the stabilization of decomposable organic matter and oxidizable inorganic matter by aerobic biological action."

The kinetic expression which will be used to describe the rate of destruction of the BOD component will now be derived.

Inspection of equations given by Grieves, Milbury, and Pipes (5) shows that the rate of formation of aerobic microorganisms, r_B, can be written as

$$r_B = \frac{dC_B}{dt} = \frac{k^M C_B x}{(K + x)} \tag{1}$$

where,

C_B = aerobic microorganism concentration

k^M = growth rate constant

x = BOD concentration

K = Miachaelis-Menton constant

If it is assumed that the organism population increase is proportional to the increase in BOD concentration, that is

$$dC_B = -Y\,dx \tag{2}$$

or

$$\frac{dC_B}{dt} = -Y\,\frac{dx}{dt} \tag{3}$$

or

$$r_B = -Y\,r_x \tag{4}$$

where

Y = a yield constant (lbs. microorganisms formed per lb. BOD consumed)

$-r_x$ = BOD reduction rate

the BOD reduction rate can be written by combining Equations (1) and (4) as

$$-r_x = \frac{k^M C_B x}{Y(K+x)} \tag{5}$$

To make the eventual optimization problem easier to solve, it is assumed that Equation (5) may be approximated by a pseudo first-order kinetic expression in terms of BOD. The approximation requires that the quotient, $k^M C_B/(Y(K+x))$, remain nearly constant throughout the portion of the lagoon which decomposes BOD, that is

$$-r_x = k' x \tag{6}$$

where

$$k' = \frac{k^M C_B}{Y(K+x)} \simeq \text{constant}$$

The validity of the assumption made to obtain Equation (6) will be verified after the optimization study has been completed.

The second idealized waste component is assumed to be a mixture of organic and inorganic compounds which pass through the lagoon without being directly oxidized or acted upon by microorganisms. This component does have an oxygen demand; however, neither direct oxidation nor attack by microorganisms while the component is in the lagoon will reduce the oxygen demand. Consequently, since the component's oxygen demand is not reduced in passing through the lagoon, it is called the nondegradeable component. The kinetics of decomposition of certain detergents are such that they are effectively nondegradeable in an aerated lagoon.

The third idealized component is assumed to be a fast reacting inorganic component with the following characteristics: It is oxidized

directly by oxygen in a relatively fast reaction. The reaction rate is assumed to be so fast that it is limited by the rate of mass transfer and the rate of addition of oxygen by the oxygenation equipment. The rate of oxidation is primarily limited by the rate of diffusion of oxygen into the waste solution. The primary example of this type of component in refinery waste water is hydrogen sulfide.

The theoretical oxygen demand (TOD) exhibited by the three components, that is, the sum of the oxygen demands of all three components, is of primary interest in this work. The chemical oxygen demand (COD) includes the biological oxygen demand and the oxygen demand of the fast reacting inorganic component.

Further description of the lagoon model can be facilitated by referring to Figure 1. The values of BOD, COD, and TOD are shown at various points along the lagoon aeration basin in the figure. At the inlet of the aeration section of the lagoon the TOD will include the oxygen demands of the BOD component, the nondegradeable component, and the fast reacting inorganic component. At the outlet of the inorganic reduction section, the TOD will be equal to the sum of the oxygen demands of the BOD and nondegradeable components, that is, in the lagoon model all of the directly oxidizable inorganics are removed in a section at the beginning of the lagoon whose volume should theoretically be determined by a diffusion limiting rate equation. In practice the aerators for this section are set as close as practically possible to the inlet.

In the lagoon model the inorganic reduction section is located at the inlet because it is desired to have as large a portion of the lagoon as possible for BOD reduction. This method of aeration is chosen because it is assumed that no aerobic bacteria can survive in the oxygen deficient

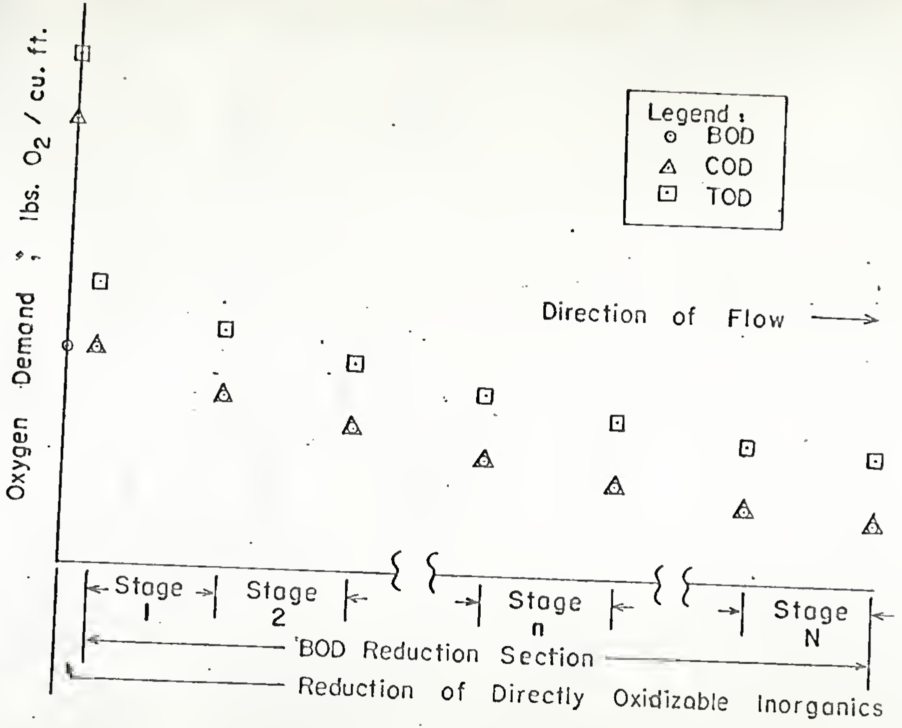

Fig. 1. BOD, COD, and TOD values which are assumed at various positions in the model of the lagoon aeration basin.

waste water which contains even a small amount of fast reacting inorganic component.

At the outlet of the aeration sections of the lagoon, the TOD will be the sum of the oxygen demands of the nondegradeable component plus the BOD of the outlet water.

2.2 BOD Material Balance

The sequential arrangement of aerators in the Sugar Creek lagoon suggested that the BOD reduction section of the lagoon could be modeled by a series of ideal backmix reactors.

A steady state BOD material balance about the nth ideal backmix (completely mixed) reactor for the biodegradable component can be written as

$$Qx^{n-1} - Qx^{n} + r_x V^{n} = 0 \tag{7}$$

where Q is the volumetric feed rate and V^n is the reactor or stage volume. The volume of the nth stage of the BOD reduction section can be calculated by combining Equations (6) and (7) to obtain

$$V^{n} = \frac{Q(x^{n-1} - x^{n})}{k^{1}\, x^{n}} \tag{8}$$

2.3 Aerator Motor Size Equation

It is assumed that Equation (8) is valid as long as a certain minimum oxygen concentration is maintained in each stage. The size of the electric motor required for maintaining this minimum oxygen concentration is directly proportional to the product of the waste water flow rate and the difference in BOD between the stage's inlet and outlet streams, that is,

$$P^{n} = \frac{Q(x^{n-1} - x^{n})}{R_o\ \ E} \tag{9}$$

where

P^n = aerator electric motor size at stage n

Q = waste water flow rate through the lagoon

x^{n-1} = oxygen demand of waste solution flowing into stage n

x^n = oxygen demand of waste solution flowing out of stage n

R_o = oxygen transfer rate constant

E = mechanical efficiency of aeration unit

2.4 Economic Model

In connection with designing an aerated lagoon, the sizes and locations of the aerators, the number of aeration stages to be used, and the lagoon volume should be determined by economic considerations.

The objective cost function used in this study takes into account both the initial equipment costs and the operating costs for the life of the equipment. A present worth objective function of the form used by Hwa (6) is used to do this. The initial costs considered are the aerator motor costs as a function of horsepower, the lagoon volume, and the costs of the aerators and supporting platforms. Only one operating cost, the electric power cost for aerator operation, is considered to have a significant effect on selecting the equipment size and lagoon volume.

Bauman (7) has proposed the following relation which can be used to obtain the initial cost of an explosion proof induction motor, $(C_I^n)_1$, as a function of the motor size, P^n,

$$(C_I^n)_1 = \beta (P^n)^\alpha \tag{10}$$

where

$$\alpha = a_1 \text{ for } 1 \leq P^n \leq 20$$

$$\alpha = a_1' \text{ for } 20 < P^n \leq 200$$

$$\beta = C_1 \text{ for } 1 \leq P^n \leq 20$$

$$\beta = C_1' \text{ for } 20 < P^n \leq 200$$

The symbols a_1, a_1', C_1, and C_1' are constants.

The initial cost of the aeration basin is taken as the sum of land real estate cost, the cost of digging the basin, and the cost of laying rock siding to prevent erosion. The unit land, digging, and rock lining costs are represented by C_2, C_3, and C_4 respectively. The aeration basin cost for each stage or aerator, $(C_I)_2$, is calculated by

$$(C_I^n)_2 = C_2 \ell w + C_3 \ell wh + C_4(\ell w + 2h\ell) \tag{11}$$

where

ℓ = length of aerator's basin

w = width of aerator's basin

h = depth of aerator's basin

To relate lagoon geometry to aerator requirements, the ratio of

$$\ell : w : h = 23:12:1 \tag{12}$$

is used. This is approximately the ratio of the dimensions of the Sugar Creek aeration basis.

Equation (12) yields

$$w = \frac{12}{23} \ell \tag{12a}$$

and

$$h = \frac{\ell}{23} \tag{12b}$$

By definition

$$V^n = \ell wh \tag{12c}$$

Substitution of Equations (12a) and (12b) into (12c) gives

$$V^n = \frac{12}{23^2} \ell^3 \tag{12d}$$

Solving Equation (12d) for ℓ^3 gives

$$\ell^3 = \frac{23^2 \, V^n}{12} \tag{12e}$$

Solving Equation (12e) for ℓ^2 gives

$$\ell^2 = \left[\frac{23^2}{12}\right]^{2/3} (V^n)^{2/3} \tag{12f}$$

Substitution of Equations (12a) and (12b) into Equation (11) gives

$$(C_I^n)_2 = C_2 \frac{12}{23} \ell^2 + C_3 \frac{12}{23^2} \ell^3 + C_4 \frac{14}{23} \ell^2 \tag{13}$$

Substitution of Equations (12e) and (12f) into Equation (13) and simplifing gives the aeration basin cost as a function of the basin volume

$$(C_I^n)_2 = C_3 V^n + C_5 (V^n)^{a_2} \tag{13a}$$

where

$$a_2 = 2/3$$

and

$$C_5 = \frac{2}{23} \left[\frac{23^2}{12}\right]^{2/3} (6C_2 + 7C_4)$$

The aerator turbine and supporting stand is assumed to have a constant initial cost, C_6, irregardless of the aerator motor size.

Since the electrical power cost is paid over a period of years, it is necessary to estimate the present worth of this money. The operating cost (C_O^n), is given by

$$(C_0^n) = \left[C_7 \, T \sum_{i=1}^{i=M} (1 + r)^{-i} \right] P^n \tag{14}$$

where

C_7 = electrical power cost (assumed constant)

T = operating hours per year

M = life of system

r = annual interest rate

Equation (14) reduces to

$$(C_0^n) = C_8 \, P^n \tag{15}$$

where

$$C_8 = C_7 T \sum_{i=1}^{M} (1 + r)^{-i} \, .$$

The total cost for each stage of the organic degradation section of the aeration basin is then written as the sum of the costs given previously:

$$G^n = (C_I^n)_1 + (C_I^n)_2 + (C_0^n) + C_6$$

or

$$G^n = C_6 + C_5 (V^n)^{a_2} + C_3 V^n + C_8 \, P^n + \beta (P^n)^{\alpha} \tag{16}$$

The objective function for minimizing the cost of the organic degradation section of the aeration basin is then defined as

$$S = \sum_{n=1}^{n=N} G^n \tag{17}$$

where

S = total cost of BOD degradation section

N = total number of aerators or stages

3.0 PROCESS OPTIMIZATION

3.1 Development of the Performance Equations

To restate our objective, the purpose of this study is to determine the volume per stage, the aerator motor size, and the number of stages required for the BOD degradation section of an aeration basin which is used to achieve a specified waste reduction while at the same time minimizing the cost of the total system.

The maximum principle is used to perform the optimization because it provides a systematic method for optimizing multi-stage processes.

In anticipation of the forms of the equations that may be used in the maximum principle solution, some of the previous equations will be combined and rearranged. Equation (8) can be rearranged to give

$$x_1^n = \frac{x_1^{n-1}}{1 + \frac{k'}{Q} V^n} \tag{18}$$

Substitution of Equation (18) into Equation (9) and rearranging gives

$$P^n = \frac{x_1^{n-1}}{R_o E \left(\frac{1}{k' V^n} + \frac{1}{Q} \right)} \tag{19}$$

The electric motor size can be eliminated from the stage cost equation by substituting Equation (19) into Equation (16)

$$G^n = C_6 + C_5 (V^n)^{a_2} + C_3 V^n + C_8 \frac{x_1^{n-1}}{R_o E \left(\frac{1}{k' V^n} + \frac{1}{Q} \right)}$$

$$+ \beta \left(\frac{x_1^{n-1}}{R\,E\,\left(\frac{1}{k'V^n} + \frac{1}{Q}\right)} \right)^{\alpha} \tag{20}$$

A new variable, x_2^n, which is equal to the sum of the costs of all the stages up to and including stage n is defined next as

$$x_2^n = x_2^{n-1} + G^n, \; x_2^0 = 0 \tag{21}$$

where

x_2^{n-1} = the sum of the costs of all stages up to stage n

G^n = the cost of stage n

By defining the stage volume of the nth stage as, θ^n, a change in notation to conform with the notation given by Fan and Wang (3) can be performed. It is hoped that the change in notation will make it easier to follow through the algorithm which is used to solve this problem. In terms of the new notation, the dependent variables which are called state variables are denoted by the letter x. The superscript n on a state variable indicates that it is the result of a decision made in the nth stage. Independent or decision variables are denoted by the Greek letter θ. The superscript n on a decision variable indicates that it is a decision which is made at the nth stage. The BOD reduction section of the aerated lagoon process can be visualized in terms of the discrete maximum principle by referring to Figure 2.

Given the values of the state variables entering a stage and the values of the decision variables at that stage, the value of a state variable leaving the stage is calculated by using its transformation

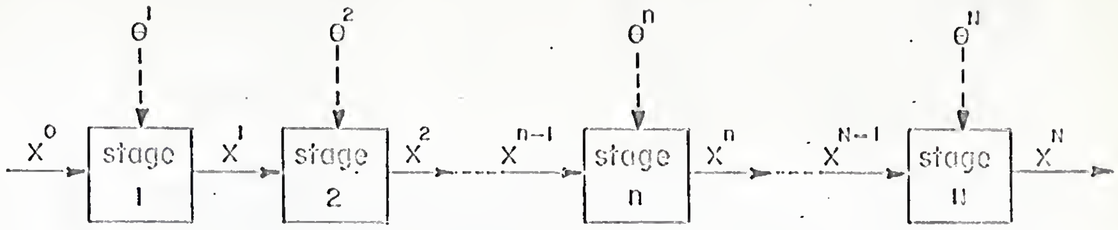

Fig. 2. The BOD reduction section of an aerated lagoon conceptionally visualized as a stagewise process.

equation (Equation (18) for x_1 and Equation (21) for x_2). In general, the transformation equations are the performance or constraint equations of the process.

3.2 Statement of the Optimization Problem

The transformation and economic equations which are needed for the optimization study can be rewritten in terms of the new notation and summarized as follows. The first number to the right of each equation gives the original equation number from which the equation is obtained. The BOD material balance equation which is the transformation equation is

$$x_1^n = \frac{x_1^{n-1}}{1 + \frac{k}{Q}\,\theta^n} = T(x_1^{n-1};\ \theta^n) \qquad (18) - (22)$$

where $T(x_1^{n-1};\ \theta^n)$ is the symbolic notation used for the transformation equation. The cost at each stage is

$$G(x_1^{n-1};\ \theta^n) = C_6 + C_5\,(\theta^n)^{a_2} + C_3\,\theta^n + C_8 \left[\frac{x_1^{n-1}}{R_o E\left(\frac{1}{k'\theta^n} + \frac{1}{Q}\right)}\right]^{\alpha} + \beta\left[\frac{x_1^{n-1}}{R_o E\left(\frac{1}{k'\theta^n} + \frac{1}{Q}\right)}\right] \qquad (20) - (23)$$

where

$\alpha = a_1$ for $1 \leq P^n \leq 20$ horsepower

$\alpha = a_1'$ for $20 \leq P^n \leq 200$ horsepower

$\beta = C_1$ for $1 \leq P^n \leq 20$ horsepower

$\beta = C_1'$ for $20 \leq 20 \leq 200$ horsepower

and the accumulated cost is

$$x_2^n = x_2^{n-1} + G(x_1^{n-1};\ \theta^n),\ x_2^0 = 0 \qquad (21) - (24)$$

The objective function which is to be minimized is

$$S = x_2^N \qquad (17) - (25)$$

The transformation equation, Equation (22), may be used to calculate the BOD concentration at the outlet of the nth stage given the inlet BOD concentration and the stage volume (the decision variable) at that stage. Equation (23) gives the cost of the nth stage in terms of the inlet BOD concentration and the stage volume. Equation (24) gives the total cost of all stages up to and including the nth stage. The total cost of the BOD reduction section of the lagoon is given by Equation (25) as the sum of the costs for each stage of the BOD reduction section.

Equations (22) through (25) are the performance equations for a one-dimensional multistage decision process which is defined by Fan and Wang (3) as a " ... process which can be completely characterized for the purpose of optimization by a single-state variable..." The stage volume is the decision or state variable for the lagoon process.

3.3 Computational Procedure - The Discrete Maximum Principle

Fan and Wang (3) have derived a necessary but not sufficient recurrence relation which can be used to calculate the optimal values of state and decision variables for many one-dimensional processes.

$$\frac{\dfrac{\partial G(x_1^{n-1};\ \theta^n)}{\partial \theta^n}}{\dfrac{\partial T(x_1^{n-1};\ \theta^n)}{\partial \theta^n}} = \frac{\dfrac{\partial G(x_1^n;\ \theta^{n+1})}{\partial \theta^{n+1}}}{\dfrac{\partial T(x_1^n;\ \theta^{n+1})}{\partial \theta^{n+1}}} \cdot \frac{\partial T(x_1^n;\ \theta^{n+1})}{\partial x_1^n} - \frac{\partial G(x_1^n;\ \theta^{n+1})}{\partial x_1^n} \qquad (26)$$

Equation (26) has the general form

$$g(x_1^{n-1};\ \theta^n) \ = \ f(x_1^n;\ \theta^{n+1}) \qquad (26a)$$

where

$$g(x_1^{n-1};\ \theta^n) \text{ and } f(x_1^n;\ \theta^{n+1})$$

are the left and right hand sides, respectively, of equation (26).

One solution procedure for a problem with fixed end points, i.e., known inlet and outlet BOD concentrations, is to start at the first stage of the process and assume a value of the decision variable for this stage, θ^1. With Equation (22), the transformation equation, x_1^1 is calculated. Next, Equation (26a) is used to calculate the optimum value of the decision variable for the next stage, θ^2, for the assumed value of θ^1. Repeated application of the transformation equation and the optimum recurrence equation for all of the stages of the process will yield a value of the outlet BOD concentration, x_1^N. If this value of x_1^N is equal to the required value, the problem has been solved. If not, another value of θ^1 is assumed and the above procedure is repeated.

The optimization calculation was carried out using the algorithm described above. However, in carrying out the computations, the electric motor size at each stage of the process, P^n, was used as the decision variable, θ^n, in place of the stage volume in order to simplify the computations. The performance equations which were used to perform the optimization are given in Appendix I. Theoretically, either the set of performance equations with stage volume as the decision variable or the set of performance equations given in Appendix I which have the stage

electric motor size as the decision variable will work equally well when performing the optimization calculations. Computer program OPT which was programmed to perform the optimization is explained in Appendix I.

4.0 RESULTS AND CONCLUSIONS

The constants which were employed in the numerical solution of the problem are given below

$a_1 = 0.53$ for $1 \leq P^n \leq 20$ horsepower

$a_1' = 1.08$ for $20 < P^n \leq 200$ horsepower

$a_2 = 0.6667$

$(BOD)_i = 0.0109025$ lb. oxygen/cu. ft. solution

$(BOD)_o = 0.0026166$ lb. oxygen/cu. ft. solution

$(COD)_i = 0.0290941$ lb. oxygen/cu. ft. solution

$(COD)_o = 0.0090958$ lb. oxygen/cu. ft. solution

Subscripts:

i = quantity in parenthesis is measured at the inlet of aeration basin

o = quantity in parenthesis is measured at the outlet of aeration basin

C_1 = \$100.00 for $1 \leq P^n \leq 20$ horsepower

C_1' = \$ 21.57 for $20 < P^n \leq 200$ horsepower

$C_2 = 1.00$ \$/sq. ft.

$C_3 = 9.259 \times 10^{-3}$ \$/cu. ft.

C_4	$= 3.333 \times 10^{-2}$	\$/sq. ft.
C_5	$= 6.764$	\$/sq. ft.
C_6	$= 5000.$	\$
C_7	$= 0.02$	\$/horsepower-hour
C_8	$= 1492.$	\$/horsepower
E	$= 0.75$	
k'	$= 0.75$	$(\text{hour})^{-1}$
M	$= 20$	years
Q	$= 45{,}120$	cu. ft./hour
r	$= 0.10$	
R_o	$= 3.2$	lb. oxygen/horsepower
T	$= 8760$	hours

$x_1^0 = (\text{BOD})_i = 0.0109025$ lb. oxygen/cu. ft. solution

Determination of the aerator motor size of the inlet section of the aeration basin (that portion of the lagoon used for directly oxidizing the idealized inorganic component) was not considered to be part of the optimization problem since the size is fixed once the waste water flow rate and the BOD's and COD's of the aeration basin inlet and outlet streams are given. The size of the inlet aerator motor, P^0, was calculated using Equation (27) and the numerical values of the constants given previously.

$$P^0 = \frac{Q}{R_o E} \left\{ \left[(\text{COD})_i - (\text{BOD})_i \right] - \left[(\text{COD})_o - (\text{BOD})_o \right] \right\} \tag{27}$$

A value of 220 horsepower was obtained for P^0.

The outlet waste solution BOD was used as a parameter in the optimization study. Three cases were considered in which it was required to obtain reductions of the inlet BOD of 76, 90, and 99 percent respectively. For the 76 percent inlet BOD reduction case, optimum policies were calculated for one, two, and three stage processes. The least expensive process is the one stage process. Optimum policies were calculated for processes composed of from one to four stages for the 90% BOD reduction case. For this case, the two stage process is the least expensive process. Similarly, optimum policies for processes with stages numbering from one to six were calculated for the 99 percent BOD reduction case. For this case, the four stage process is least expensive. The results for the cases of 76, 90, and 99 percent inlet BOD reduction are summarized in Tables 1, 2, and 3, respectively. The optimum process for each case is given in Table 4.

Since the maximum principle does not guarantee a minimum, it is necessary to do simulation studies with a mathematical model of the lagoon to insure that a minimum has been found. Simulation studies were performed with computer program SIM which was programmed to simulate the process taking place in the lagoon aeration basin based upon performance Equations (22) through (25).

The results of the simulation studies for the 99 percent BOD reduction case are given in Table 5. Three cases given in Table 5 are slightly less expensive than the optimum process found by the maximum principle solution. These less expensive processes resulted because of discretization error which was introduced by computer program OPT during the optimization calculations. The optimum process can be computed more accurately, however, the improvement obtained in the total process cost will not be significant enough to justify additional computations. In the fourth simulation study

given in Table 5, the aerator motor size of each stage was obtained by rounding off the optimum aerator motor sizes of the first three stages to integer sizes as would be required should it be desired to purchase the aerator motors. The total process cost of the fourth simulation study is not significantly more than the total cost of the optimum process.

For the model used in this study, several trends in the relationships among the variables are apparent upon inspection of Tables 1, 2, 3, and 4. At a given conversion, the optimum total volume required to achieve this conversion decreases as the number of stages increases. Both the total cost and the optimum number of stages for the BOD reduction section of the lagoon increases as the percentage BOD conversion increases. For all of the multi-stage processes, tapered aeration as discussed by Sawyer (8), is the best policy. This is due to the fact that the rate of BOD reduction decreases as the treatment progresses.

The discrete maximum principle shows promise as a useful technique for the optimization of aerated lagoons which are operated in a stage-wise fashion. The use of this technique with the aerated lagoon model used in this work predicts, as was expected, that tapered aeration is the optimal operating policy for multi-stage lagoons.

For all of the optimal cases, the total aeration volume in each case is an order of magnitude smaller than the Sugar Creek aeration basin volume (about 1,700,000 cu. ft.). This may be partially due to the value of the first order reaction rate constant, k', which was used in the numerical computations. Too large a value of k' would account for the smaller optimal aeration volume calculated. An improved model with calculations to support this belief is given below.

Table 1. Optimum Policies for 76 Percent BOD Reduction.

Process	Stage Number n	Inlet BOD $\frac{\text{lbs. } O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) P^n	Cost of Stage (\$) $G(x_1^{n-1}; \theta^n)$
One Stage	1	0.010902	190,674	155.8	266,595
Two Stage	1	0.010902	60,892	103.1	173,037
	2	0.005418	64,449	52.6	96,609
		Total	125,341	155.7	269,647
Three Stage	1	0.010902	32,229	71.5	120,967
	2	0.007099	37,429	51.1	90,777
	3	0.004376	40,490	33.0	63,661
		Total	110,149	155.7	275,406

Table 2. Optimum Policies for 90 Percent BOD Reduction.

Process	Stage Number n	Inlet BOD $\frac{lbs.\ O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) P^n	Cost of Stage ($) $G(x_1^{n-1};\ \theta^n)$
One Stage	1	0.010902	542,638	184.5	336,265
Two Stage	1	0.010902	128,498	139.6	236,109
	2	0.003476	132,096	44.9	92,059
		Total	260,594	184.5	328,169
Three Stage	1	0.010902	66,617	107.7	180,754
	2	0.005173	70,131	52.3	96,784
	3	0.002388	71,771	24.4	54,464
		Total	208,520	184.4	332,005
Four Stage	1	0.010902	39,632	81.1	137,141
	2	0.006572	44,108	52.3	93,363
	3	0.003792	46,669	31.1	61,536
	4	0.002135	59,731	19.9	46,210
		Total	190,142	184.7	338,251

Table 3. Optimum Policies for 99 Percent BOD Reduction.

Process	Stage Number n	Inlet BOD $\frac{\text{lbs. } O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) P^n	Cost of Stage (\$) $G(x_1^{n-1}; \theta^n)$
One Stage	1	0.010902	5,944,204	202.9	591,349
Two Stage	1	0.010902	536,801	184.3	335,582
	2	0.001098	547,088	18.6	83,539
		Total	1,083,889	202.9	419,122
Three Stage	1	0.010902	216,572	160.4	275,843
	2	0.002370	220,049	34.9	84,884
	3	0.000508	221,935	7.5	43,367
		Total	658,557	202.9	404,094
Four Stage	1	0.010902	124,818	138.3	233,761
	2	0.003545	128,349	45.3	92,420
	3	0.001131	134,779	14.7	46,381
	4	0.000349	133,429	4.5	30,869
		Total	521,376	202.9	403,433
Five Stage	1	0.010902	86,526	120.9	203,193
	2	0.004471	90,057	50.3	96,076
	3	0.001790	91,507	20.3	50,436
	4	0.000710	93,924	8.1	32,290
	5	0.000277	92,845	3.1	24,631
		Total	454,862	202.9	406,626

Table 3. (Continued)

Process	Stage Number n	Inlet BOD $\frac{\text{lbs. } O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) p^n	Cost of Stage (\$) $G(x_1^{n-1};\ \theta^n)$
Six Stage	1	0.010902	62,946	104.8	175,884
	2	0.005327	66,712	52.6	96,857
	3	0.002526	68,399	25.2	55,340
	4	0.001182	74,455	12.2	36,372
	5	0.000528	72,994	5.4	25,856
	6	0.000238	72,049	2.4	21,187
		Total	417,558	202.9	411,498

Table 4. Minimum Cost Aerated Lagoon Processes for 76, 90 and 99 Percent Reduction of BOD.

Process	Stage Number n	Inlet BOD $\frac{\text{lbs. } O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) P^n	Cost of Stage (\$) $G(x_1^{n-1};\ \theta^n)$	BOD Reduction (%)
One	1	0.010902	190,674	155.8	266,595	76.0
Two Stage	1	0.010902	128,498	139.6	236,109	
	2	0.003476	132,096	44.9	92,059	90.0
		Total	260,594	184.5	328,169	
Four Stage	1	0.010902	124,818	138.3	233,761	
	2	0.003545	128,349	45.3	92,420	
	3	0.001131	134,779	14.7	46,381	
	4	0.000349	133,429	4.5	30,869	99.0
		Total	521,376	202.9	403,433	

Table 5. Process Simulation Studies Made for the 99 Percent BOD Reduction Case.

Process	Stage Number	Inlet BOD $\frac{lbs.\ O_2}{ft^3}$	Stage Volume (ft^3)	Motor Size (H.P.)	Cost of Stage ($)
	n	x_1^{n-1}	θ^n	P^n	$G(x_1^{n-1};\ \theta^n)$
Four Stages	1	0.010903	124,800	138.3	233,759
	2	0.003546	128,444	45.4	92,456
	3	0.001131	134,743	14.7	46,368
	4	0.000349	131,231	4.5	30,617
		Total	519,220	202.9	403,202
Four Stages	1	0.010903	125,078	138.4	233,939
	2	0.003540	129,949	45.5	92,756
	3	0.001120	142,158	14.8	47,234
	4	0.000333	122,482	4.2	29,296
		Total	519,669	202.9	403,226
Four Stages	1	0.010903	125,357	138.5	234,120
	2	0.003535	131,483	45.6	93,059
	3	0.001109	150,320	14.9	48,158
	4	0.000317	113,734	3.9	27,955
		Total	520,895	202.9	403,293
Four Stages	1	0.010903	129,641	140.0	236,832
	2	0.003455	135,603	45.0	92,547
	3	0.001061	181,821	15.0	51,187
	4	0.000263	84,571	2.9	23,317
		Total	531,636	202.9	403,885

Table 5. (Continued)

Process	Stage Number n	Inlet BOD $\frac{\text{lbs. } O_2}{ft^3}$ x_1^{n-1}	Stage Volume (ft^3) θ^n	Motor Size (H.P.) P^n	Cost of Stage (\$) $G(x_1^{n-1};\ \theta^n)$
Four Stages	1	0.010903	2,375,218	199.9	452,169
	2	0.000269	14,806	1.0	10,807
	3	0.000216	19,640	1.0	11,697
	4	0.000162	29,162	1.0	13,270
		Total	2,438,828	202.9	487,945
Four Stages	1	0.010903	19,822	50.8	87,410
	2	0.008200	29,478	50.7	88,848
	3	0.005503	57,804	50.7	92,768
	4	0.002806	1,478,544	50.7	183,603
		Total	1,585,649	202.9	452,630
Four Stages	1	0.010903	1,350	4.5	12,772
	2	0.010663	4,760	14.7	29,300
	3	0.009881	19,457	45.4	79,120
	4	0.007466	4,033,146	138.3	424,451
		Total	4,058,714	202.9	545,644

An improved lagoon model can be derived by making material balances for both the microorganisms and the BOD component around each of the ideal backmix stages of the lagoon. The material balances for the microorganisms and the BOD component are given by Equations (28) and (29), respectively.

$$C_B^{n-1} - C_B^n + \left[\frac{k^M x_1^n C_B^n}{K + x_1^n} - k_D C_B^n \right] \theta^n = 0 \tag{28}$$

$$x_1^{n-1} - X_1^n - \left[\frac{k^M x_1^n C_B^n}{Y(K + x_1^n)} \right] \theta^n = 0 \tag{29}$$

where

C_B^n = the microorganism concentration in the nth stage

k_D = the endogeneous respiration rate constant

θ^n = V^n/Q

If the power requirement for each stage is calculated by Equation (9) and if the total stage cost is calculated by Equation (16), then for a two stage process a one dimensional search can be used to determine the values of P^n and V^n which will give the lowest aeration basin cost.

A one dimensional exhaustive search was performed for 76, 90 and 99 percent BOD reduction. The search variable used was x_1^1, the first stage outlet BOD. With x_1^1 known, and the constants given below, C_B^1, C_B^2, θ^1, θ^2, P^1, P^2 and the total cost of the process may be calculated. The exhaustive search procedure was used to calculate the total process cost at a series of x_1^1 values taken at intervals between x_1^0 and x_1^2. The process with the x_1^1 value which gave the lowest total cost was selected as the optimum. The above procedure was programmed for use on an IBM 1410 computer.

The name of the computer program which was used to perform the one dimensional search calculations is SEARCH. Program SEARCH is described in Appendix III. The numerical values of the constants used in program SEARCH are as follows:

$$C_B^0 = 0.0 \text{ ppm} \qquad K = 100 \text{ ppm}$$

$$k^M = 0.1 \text{ hr}^{-1} \qquad x_1^0 = 175 \text{ ppm}$$

$$k_D = 0.002 \text{ hr}^{-1} \qquad Y = 0.5 \frac{\text{lbs. microorganisms formed}}{\text{lb. BOD consumed}}$$

The optimum results obtained from the exhaustive search for the three cases are given in Table 6.

For the case of 76 per cent BOD reduction a one stage process was optimum. If the values of C_R^1, k, K, and x_1^1 for the optimum 76 per cent BOD reduction case are used in the equation which defines k', a value of k' equal to 0.0437 results. This new k' value when used in computer program SIM, the lagoon process simulation program, gives a lagoon volume of 3,272,449. cu. ft. and a total cost of \$421,824. for the one stage 76 per cent BOD reduction case. This lagoon volume is of the same order of magnitude as the one at Sugar Creek.

The minimum cost solution for 90 per cent BOD reduction is a two stage process. Using this solution and the defining equation for k', values of 0.0433 and 0.0619 were obtained for the first and second stages, respectively. These numbers indicate that the assumption of a constant k' value throughout the process is not reasonable. In the next section of this report, a lagoon aeration basin model is proposed which gives a more detailed account of the processes which occur in the aeration basin. Optimization of this model should give more realistic answers.

Table 6. Results of the One Dimensional Search for the Two Stage Lagoon Aeration Basin Which Is Modeled by Equations (28) and (29).

	Stage Number			Total for all stages	% Conversion of BOD
	0	1	2		
C (ppm)	0.00	62.00			
X (ppm)	175.00	42.00			76
V (ft^3)		1,635,980.00			
P (H.P.)		155.76			
0 (hr.)		36.26			
G ($)		351,432.00			
C (ppm)	0.00	61.73	72.64		
X (ppm)	175.00	42.70	17.50		90
V (ft^3)		1,615,750.00	525,492.00	2,141,240.00	
P (H.P.)		154.94	29.51	184.45	
0 (hr.)		35.81	11.65	47.46	
G ($)		349,218.00	98,772.80	447,991.00	

Table 6. (cont'd.)

	Stage Number			Total for all stages	% Conversion of BOD
	0	1	2		
C (ppm)	0.00	68.13	66.88		
X (ppm)	175.00	17.00	1.75		99
V (ft^3)		3,600,740.00	2,990,560.00	6,591,310.00	
P (H.P.)		185.04	17.86	202.90	
O (hr.)		79.81	66.28	146.09	
G ($)		479,323.00	200,204.00	679,527.00	

5.0 PROPOSED FUTURE WORK

More accurate performance equations for the aerated lagoon process can be obtained by quantitatively considering the primary components which are present. The model used in the present optimization study assumed that the BOD reduction rate could be approximated by a pseudo first order kinetic expression written in terms of BOD concentration. As shown in the last section this assumption is not valid since the pseudo rate constant, k', varies appreciably throughout the process. Furthermore, the effects of oxygen mass transfer rate and oxygen concentration upon reaction rates need to be considered in detail.

The kinetic and mass transfer expressions for the aerated lagoon model which is proposed for future work are derived in the following paragraphs.

Monod (9) first used a model to express the rate of growth of microorganisms which represents the rate as being proportional to the nutrient or substrate concentration at low nutrient concentrations but reaching a limiting value at high nutrient concentrations:

$$r_B = k^M x_2^n \left(\frac{x_1^n}{K + x_1^n}\right) \tag{30}$$

where,

r_B = rate of microorganism growth

k^M = growth rate constant

x_1^n = nutrient concentration

x_2^n = microorganism concentration

K = Michaelis-Menten constant

This model assumes that sufficient oxygen is present to satisfy the needs of the organisms.

The problem of maintaining adequate oxygen concentrations has been recognized. However, little work has been done to determine adequate oxygen levels even for pure cultures.

As a means of taking into account the effect of oxygen concentration upon the rate of formation of organisms, a modification of Equation (30) is used in this work.

$$r_B = k^M x_2^n \left(\frac{x_1^n}{K + x_1^n}\right) \left(\frac{x_3^n}{K_0 + x_3^n}\right) \tag{31}$$

where,

x_3^n = oxygen concentration

K_0 = a constant similar to the Michaelis-Menten constant

In aeration systems, microorganisms are lost through a process known as endogenous respiration. In effect, they are consumed or oxidized by other microorganisms. McKinney (10) used a first order kinetic equation to express the endogenous respiration rate, r_e. The equation can be written as

$$r_e = k_D x_2^n \tag{32}$$

where k_D is the endogenous respiration rate constant.

By analogy to Equation (31), the factor, $(x_3^n/(K_0 + x_3^n))$, is used in this work to take into account the effect of low oxygen concentrations upon the endogenous respiration rate

$$r_e = k_D x_2^n \left(\frac{x_3^n}{K_0 + x_3^n}\right) \tag{33}$$

If it is assumed that the microorganism population increase is proportional to the decrease in nutrient concentration, then the nutrient reduction rate, r_x , can be given by

$$r_x = - \frac{r_B}{Y} \tag{34}$$

where Y is the yield factor.

The oxygen consumption rate, r_0, is equal to the sum of the endogenous respiration and nutrient reduction rates

$$r_0 = r_e + r_x \tag{35}$$

Oxygen is supplied to a stage of the aeration basin by the aerator. Since aeration is a mass transfer operation, it is taken into account by letting the mass transfer rate per unit of stage volume be equal to the product of a combined overall liquid phase mass transfer coefficient and interfacial area, K_La, and a driving force which is given by the difference between the oxygen saturation concentration and the oxygen concentration in the stage, $(x_3^s - x_3^n)$. To calculate oxygen input rate as a function of aerator horsepower, the mass transfer coefficient is written as a function of the aerator horsepower, shaft speed, and the stage volume as follows:

$$K_La = K_c \left(\frac{P^n}{R}\right)^c (V^n)^{-b} \tag{36}$$

where

b, c, and K_c are constants

and

P^n = aerator horsepower

R = aerator shaft speed

V^n = stage volume

Equation (36) was suggested by the investigations of Blakebrough and Sambamurthy (11), which were conducted in batch laboratory fermentors.

A nutrient, or BOD, mass balance for an ideal backmix stage of the nutrient degradation section of the lagoon may be written as

$$V^n \frac{dx_1^n}{dt} = Q\, x_1^{n-1} + V^n\, r_x - Q\, x_1^n \tag{37}$$

The terms of Equation (37) take into account accumulation of nutrient in the stage, nutrient washing into the stage, nutrient degradation in the stage, and nutrient washing out of the stage, respectively.

Similar mass balances for microorganisms and oxygen are given by Equations (38) and (39), respectively.

$$V^n \frac{dx_2^n}{dt} = Q\, x_2^{n-1} + V^n\, r_B - Q\, x_2^n \tag{38}$$

$$V^n \frac{dx_3^n}{dt} = Q\, x_3^{n-1} + V^n \left[K_L a\, (x_3^s - x_3^n) + r_0 \right] - Q\, x_3^n \tag{39}$$

The performance equations for the model which is proposed for future work are summarized in Table 7. When used in conjunction with a stage cost equation such as Equation (23), this model can be used to obtain more accurate values of the independent variables of the optimization study.

Table 7. Performance Equations for Model Proposed for Future Work.

Rate Term (ppm/hr)	Component		
	Nutrient	Microorganisms	Oxygen
Accumulation	$V^n \frac{dx_1^n}{dt}$	$V^n \frac{dx_2^n}{dt}$	$V^n \frac{dx_3^n}{dt}$
Input	$Q\ x_1^{n-1}$	$Q\ x_2^{n-1}$	$Q\ x_3^{n-1} + V^n K_L a(x_3^s - x_3^n)$
Gain by Reaction	-----	$V^n\ r_B$	-----
Loss by Reaction	$V^n\ r_x$	$V^n\ r_e$	$V^n\ r_0$
Outflow	$Q\ x_1^n$	$Q\ x_2^n$	$Q\ x_3^n$

6.0 REFERENCES

1. Stroud, P. W., L. V. Sorg, and J. C. Lamkin, "The First Large Scale Industrial Waste Treatment Plant on the Missouri River", Proc. 18th Indust. Waste Conference, May 1963, Purdue University (1964).

2. Burkhead, C. E., M. S. Thesis, University of Kansas, Lawrence, Kansas, (1963).

3. Fan, L. T., and C. S. Wang, "The Discrete Maximum Principle-A Study of Multi-Stage Systems Optimization", Wiley, New York (1964).

4. Eckenfelder, W. W., and D. J. O'Connor, "Biological Waste Treatment", Pergamon Press, New York (1961).

5. Grieves, R. B., Milbury, W. F., and W. O. Pipes, Jr., "A Mixing Model for Activated Sludge", Proc. 18th Indust. Waste Conference, May 1963, Purdue University (1964), and Journal WPCF, 36, 640, (1964).

6. Hwa, C. S., "Mathematical Formulation and Optimization of Exchanger Networks Using Separable Programming", A.I.C.L.E.-I Chem. E. Symposium, Series 4, The Institution of Chemical Engineers, 16 Belgrave Square, London, S. W. 1, England (1965).

7. Bauman, H. C., "Fundamentals of Cost Engineering in the Chemical Industry", Reinhold, New York (1964).

8. Sawyer, C. N., "Milestones in the Development of the Activated Sludge Process", Journal, Water Pollution Control Federation, 37, No. 2, 151-162 (1965).

9. Monod, J., "The Growth of Bacteria Cultures", Annual Review of Microbiology, 3, 371 (1949).

10. McKinney, R. E., "Mathematics of Complete Mixing Activated Sludge", Proc. American Soc. of Civil Engr., SA 3, 91 (1962).

7.0 NOMENCLATURE

a_1, a_1'	Constants used in Equation (10)
b	Constant used in Equation (30)
c	Constant used in Equation (30)
BOD	Biological oxygen demand
C_B	Aerobic microorganism concentration
$(C_I^n)_1$	Initial investment cost of aerator motor
$(C_0^n)2$	Cost of aeration basin
(C_0^n)	Present worth of electrical power cost which is paid over a period of years
COD	Chemical oxygen demand
C_1, C_1'	Constants used in Equation (10)
C_2	Unit land cost of aeration basin
C_3	Unit digging cost of aeration basin
C_4	Unit cost for rock lining of aeration basin
C_5	Constant defined as $\frac{2}{23} \left[\frac{23^2}{12}\right]^{2/3} (6C_2 + 7C_4)$
C_6	Initial cost of aerator turbine and supporting stand
C_7	Electrical power cost
C_8	Constant defined as $C_7\, T \sum_{i=1}^{M} (1 + r)^{-i}$
E	Mechanical efficiency of aeration unit
$f\,(x_1^{n-1};\, \theta^{n+1})$	Right hand side of optimal recurrence relation, Equation (26)

$g\ (x_1^n;\ \theta^n)$	Left hand side of optimal recurrence reaction, Equation (26)
$G^n, G(x_1^{n-1};\ \theta^n)$	Total cost of n-th aeration basin
h	Aeration basin depth
k'	Pseudo first order reaction rate constant
k^M	Microorganism growth rate constant
k_D	Endogenous respiration rate constant
K	Michaelis - Menton constant
K_c	Constant used in Equation (30)
$K_L a$	Combined overall liquid phase mass transfer and interfacial area
K_0	Constant similar to K
ℓ	Length of aeration basin
M	Useful life of aeration basin
n	Superscript denoting the n-th stage of the aeration basin
N	Superscript denoting the last stage of the aeration basin
p^n	n-th stage aerator motor size
p^0	Aerator motor size for inlet section of aeration basin which directly oxidizes the idealized inorganic component.
Q	Waste water flow rate to aeration basin
r	Simple interest rate
r_B	Aerobic microorganism formation rate
r_e	Endogenous respiration rate

r_0	Oxygen utilization rate
r_x	BOD, i.e., nutrient reduction rate
R	Aerator shaft speed
R_o	Oxygen transfer rate constant
S	Total cost of BOD degradation section at aeration basin
t	Time
T	Aeration basin operating hours per year
V^n	Volume of n-th aeration basin stage
w	Width of lagoon
x	BOD concentration
x^{n-1}, x^n	BOD concentration in (n-1)-th and n-th stage of aeration basin, respectively
x_1^N	BOD concentration in waste water leaving aeration basin
x_1^0	BOD concentration in waste water flowing into aeration basin
x_2^n	Sum of costs of all stages of the aeration basin up to and including the n-th stage. Used in Equations (21) and (24)
x_2^n	Microorganism concentration in n-th stage of aeration basin. Used in Equation (31) and thereafter.
x_3^n	Oxygen concentration in n-th stage of aeration basin.
x_3^s	Oxygen saturation concentration
Y	Yield constant $\left(\frac{\text{lbs. microorganisms formed}}{\text{lb. BOD consumed}}\right)$

Greek Letters

α	The power to which a term in Equation (23) is raised
β	The coefficient of a term used in Equation (23)
θ^n	Volume of n th stage as used in Equations (22) through (24)
θ^n	Holding time of n th stage of aeration basin (V^n/Q). Used in Equations (28) and (29).
θ^{n+1}	Aerator motor size. Used in Appendix I.

APPENDIX I

EXPLANATION OF LAGOON OPTIMIZATION COMPUTER PROGRAM - OPT AND SUBROUTINE - DMP

To simplify programming of OPT, the computer program which was used to perform the optimization calculations, different performance equations were used in place of the performance equations given in Section 3.2. The transformation equation which was used in OPT is obtained by rearranging Equation (9) and defining the electric motor size as the n th stage decision variable, θ^n.

$$x_1^n = x_1^{n-1} - \theta^n R_o E/Q = T(x_1^{n-1}; \theta^n) \qquad \text{(I-1)}$$

where

x_1^{n-1} = oxygen demand of the waste solution flowing into stage n

x_1^n = oxygen demand of the waste water solution flowing out of stage n

θ^n = aerator electric motor size at stage n

R_o = oxygen transfer rate constant

E = mechanical efficiency of aeration unit

Q = waste water flow rate into aeration basin

The stage volume required for stage n, V^n, is calculated by combining Equations (8) and (9) so that the outlet BOD is eliminated from the resulting equation which is given below.

$$V^n = \frac{Q\,\theta^n}{k'\left(\dfrac{Qx_1^{n-1}}{R_oE} - \theta^n\right)} \qquad \text{(I-2)}$$

The cost of stage n is calculated using Equation (I-3) which is obtained by substituting Equation (I-2) into Equation (16).

$$G(x_1^{n-1};\, \theta^n)$$

$$= C_6 + C_8\,\theta^n + \beta(\theta^n)^{\alpha} + C_3 \left[\frac{Q\,\theta^n}{k'\left(\dfrac{Qx_1^{n-1}}{R_oE} - \theta^n\right)}\right]$$

$$+ C_5 \left[\frac{Q\,\theta^n}{k'\left(\dfrac{Qx_1^{n-1}}{R_oE} - \theta^n\right)}\right]^{a_2} \qquad \text{(I-3)}$$

where

$$\alpha = a_1 \text{ when } 1 \leq \theta^n \leq 20$$

$$\alpha = a_1' \text{ when } 20 < \theta^n \leq 200$$

$$\beta = C_1 \text{ when } 1 \leq \theta^n \leq 20$$

$$\beta = C_1' \text{ when } 20 < \theta^n \leq 200$$

The accumulated cost of the process is

$$x_2^n = x_2^{n-1} + G(x_1^{n-1};\, \theta^n),\; x_2^0 = 0 \qquad \text{(I-4)}$$

The objective function which is to be minimized is

$$S = x_2^N \qquad \text{(I-5)}$$

The transformation equation, Equation (I-1), may be used to calculate the BOD concentration at the outlet of the nth stage given the inlet BOD concentration and the electric motor size of the stage (the decision

variable). The volume of stage n may be calculated using Equation (I-2). The cost of stage n may be calculated using Equation (I-3). Equation (I-4) gives the total cost of all stages up to and including stage n. The total cost of the BOD reduction section of the lagoon is given by Equation (I-5). Equations (I-1) through (I-5) are the performance equations which were used in programming computed program OPT.

For the transformation and stage cost equations given in this Appendix, the left hand side of the necessary optimal recurrence equation, Equation (26), is given by Equation (i-6). The right hand side of Equation (26) is given by Equation (I-7).

$$g(x_1^{n-1};\ \theta^n)$$

$$= \left[C_3 + a_2 C_5 \left[\frac{Q\ \theta^n}{k'\left(\frac{Qx_1^{n-1}}{R_o E} - \theta^n\right)} \right]^{a_2 - 1} \right] \left[\frac{\frac{Q^2}{k' R_o E}\, x_1^{n-1}}{\left(\frac{Qx_1^{n-1}}{R_o E} - \theta^n\right)^2} \right] + \alpha\beta(\theta^n)^{\alpha-1} + C_8 \tag{I-6}$$

$$f(x_1^n;\ \theta^{n+1})$$

$$= \left[C_3 + a_2 C_5 \left[\frac{Q\ \theta^{n+1}}{k'\left(\frac{Qx_1^n}{R_o E} - \theta^{n+1}\right)} \right]^{a_2 - 1} \right] \left[\frac{\frac{Q^2}{k' R_o E}\, x_1^n}{\left(\frac{Qx_1^n}{R_o E} - \theta^{n+1}\right)^2} \right] \left(1 - \frac{R_o E\ \theta^{n+1}}{Q}\right)$$

$$+ \alpha\beta(\theta^{n+1})^{\alpha-1} + C_8 \tag{I-7}$$

where

$\alpha = a_1$ for $1 \leq \theta^n \leq 20$ or $1 \leq \theta^{n+1} \leq 20$

$\alpha = a_1'$ for $20 < \theta^n \leq 200$ or $20 < \theta^{n+1} \leq 200$

$\beta = C_1$ for $1 \leq \theta^n \leq 20$ or $1 \leq \theta^{n+1} \leq 20$

$\beta = C_1'$ for $20 < \theta^n \leq 200$ or $20 < \theta^{n+1} \leq 200$

The values of α and β used in this problem change abruptly at θ^n or θ^{n+1} equal to twenty because the aerator motor cost versus size relation, which was used in this optimization study, has a discontinuity at a value of twenty horsepower. This abrupt change in α and β causes the functions g $(x_1^{n-1}; \theta^n)$ and f $(x_1^n; \theta^{n+1})$, when plotted versus θ^n and θ^{n+1}, respectively, to be discontinuous at θ^n or θ^{n+1} equal to twenty. The discontinuity in the f $(x_1^n; \theta^{n+1})$ versus θ^{n+1} curve is shown in Figure I-1 for a typical f $(x_1^n; \theta^{n+1})$ function. Even though the derivation of the necessary recurrence equation, Equation (26), requires that the functions g $(x_1^{n-1}; \theta^n)$ and f $(x_1^n; \theta^{n+1})$ be continuous over the range of θ^n and θ^{n+1} values encountered in the process, it was found that the maximum principle may still be used when the objective function is piecewise smooth. The necessary recurrence equation is valid over each piecewise smooth segment of the f $(x_1^n; \theta^{n+1})$ versus θ^{n+1} curve. The search technique which is used to find θ^{n+1} is modified to take into account the discontinuity. This procedure allows the search technique to search the whole range of θ^{n+1} values. The search technique and the modification which was used is discussed in the following paragraphs which describe the computer program OPT.

Computer program OPT can be easily described by reviewing the computational procedure given in Section 3.3. By starting at the first stage of the process and assuming a value of θ^1, a fixed value of g $(x_1^o; \theta^1)$ may be calculated using Equation (I-6). The BOD at the outlet of stage one, x_1^1, may be calculated by using the transformation equation, Equation (I-1). A value of θ^2 may be calculated by using the necessary recurrence equation which, for this problem, is obtained by equating Equations (I-6)

and (I-7). The necessary recurrence equation has the same form as Equation (26a).

Due to the complexity of the right hand side of the necessary recurrence equation, i.e., the function f $(x_1^n; \theta^{n+1})$, it is expedient to use a search technique to solve for θ^2. Similarly, a value of the decision variable for each stage of the process may be calculated by alternate, repeated applications of the transformation equation and searching the necessary recurrence equation. If the value of the outlet BOD, x_1^N, is not equal to the required value, the above procedure is repeated by assuming a new value for θ^1. If the value of x_1^N is sufficiently close to the required value, no further calculations are required.

Computer program OPT was programmed to carry out the computational procedure given above. When work was first commensed on OPT, it was assumed that only one value of θ^{n+1} would be found when searching the necessary recurrence equation over the range of θ^{n+1} values which are feasible for the process. This situation is indicated by the dashed line in Figure I-1. Several trials using the Bolzano search technique, which is described by Stanton (1), revealed that the computation time for this technique was not excessive. Consequently, the Bolzano technique was selected for use in OPT to search for θ^{n+1}.

After the Bolzano technique was incorporated into OPT, very small values of the state variables were obtained for some cases tried. Further investigation revealed that two values of θ^{n+1} satisfy the necessary recurrence equation. This situation is shown by the solid curve for $f(x_1^n; \theta^{n+1})$ on Figure I-1. Optimum processes were not obtained using the smaller θ^{n+1} values. Therefore, the Bolzano technique was made to converge to the larger θ^{n+1} values. Convergence to the larger θ^{n+1} values was assured by starting the Bolzano technique on the portion of the $f(x_1^n; \theta^{n+1})$

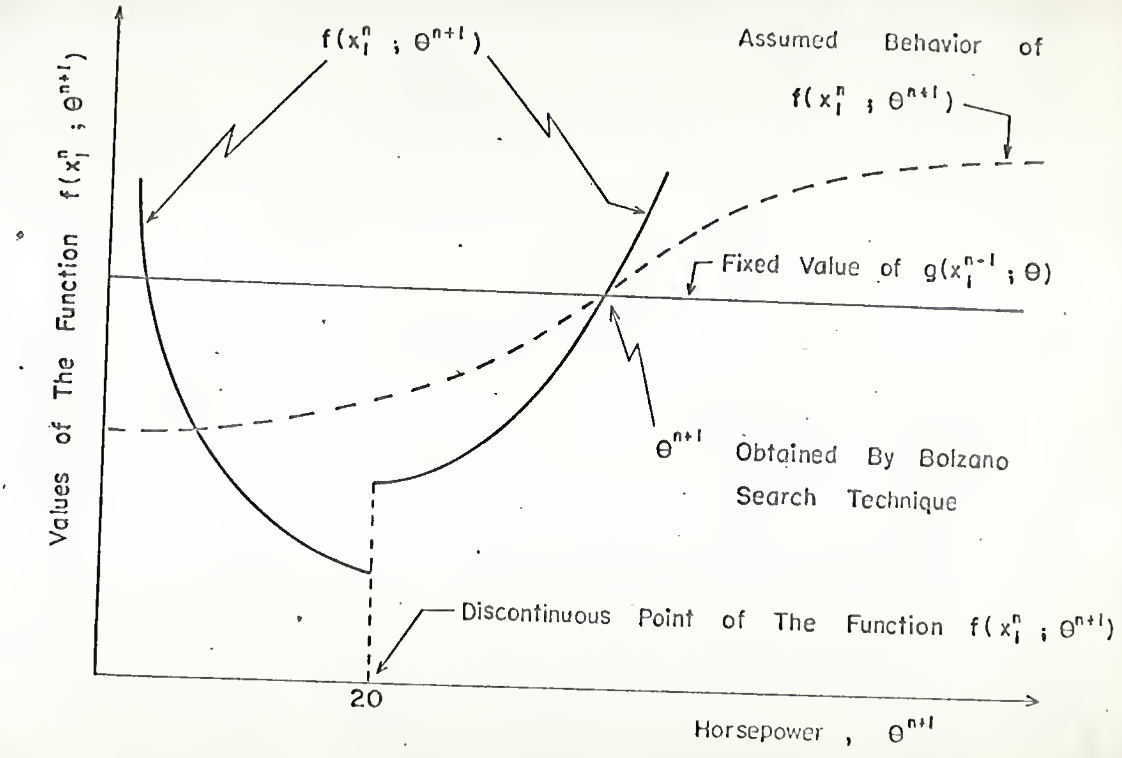

Fig. I-1. A typical plot of the function $f(x_1^n; \theta^{n+2})$ versus θ^{n+1}.

versus θ^{n+1} curve which has a positive slope.

It was discovered that when the θ^{n+1} value being searched for was near the discontinuous point of the $f(x_1^n;\ \theta^{n+1})$ versus θ^{n+1} curve, the Bolzano technique failed to converge to a θ^{n+1} value. The reason for the failure of the Bolzano technique to converge to a θ^{n+1} value was not investigated because a simple solution for this difficulty was found. It was observed that the Bolzano technique converged to a θ^{n+1} value at positions away from the discontinuity in only a few trials. Consequently, a limit was placed on the number of trials which were permitted for the Bolzano technique to search for a θ^{n+1}. In the event that the number of trials used by the Bolzano technique exceeded the limit, i.e., in the event that the technique failed to converge, θ^{n+1} was set equal to the θ^{n+1} value just to the left hand side of the discontinuity because this θ^{n+1} value gives a lower aerator motor cost than the θ^{n+1} value just to the right of the discontinuity.

This work has shown that it is possible to extend the use of the discrete maximum principle to solve optimization problems which have piecewise continuous objective functions. Rosenbrock and Storey (2) have pointed out that piecewise continuous cost versus size curves are frequently encountered when dealing with power equipment.

Subroutine DMP is used with program OPT. Subroutine DMP consists of several defined functions and it is used to reduce the size of OPT.

Table I-1 explains the symbols used in OPT and DMP. Logic diagrams for OPT and DMP are given in Figures I-2 and I-3, respectively. Program OPT and subroutine DMP are listed in Table I-2.

Table I-1. Explanation of Computer Program Variables for Program OPT and Subroutine DMP

Symbol	Explanation
A	Current sum of $G(x_1^n;\ \theta^{n+1})$ values,
A1	a_1 on a_1'
A2	a_2
AC1	C_6
AC2	C_8
AC3	C_1
AC4	C_5
AC5	C_3
B	Current sum of P^n values
B1	Q/R_oE
B2	Q/k'
C	Current sum of V^n values
DGDT	$\partial G(x_1^n;\ \theta^{n+1}) / \partial \theta^{n+1}$
DGDX	$\partial G(x_1^n;\ \theta^{n+1}) / \partial x_1^n$
DT	Incremental change in θ^{n+1}
DTDT	$\partial T(x_1^n;\ \theta^{n+1}) / \partial \theta^{n+1}$
DTDX	$\partial T(x_1^n;\ \theta^{n+1}) / \partial x_1^n$
DUDX	$\partial V^n / \partial x_1^n$
D2GDT	$\partial^2 G(x_1^n;\ \theta^{n+1}) / (\partial \theta^{n+1})^2$
D2UDT	$\partial^2 V^n / (\partial \theta^{n+1})^2$
E	Allowable error
EFF	E
F	A defined function used to simplify the computer program

Table I-1. (con't)

Symbol	Explanation
G	$G(x_1^n;\ \theta^{n+1})$
GL	Left hand side of optimal recurrence equation
GR	Right hand side of optimal recurrence equation
G1,G2,G3	Defined functions used to simplify calculation at D2GDT
I	A counter used as the stage number
II	A counter used to determine the number at times the Bolzano technique is applied at each stage.
IT	Maximum number at times Bolzano technique is applied at each stage
J	A counter used in the logic steps of the Bolzano technique
K	Number of stages in process
Q	Q
QA	Multiplication factor used to obtain a positive value of D2GDT.
RK	k'
S	A dummy variable used to sum $G(x_1^n;\ \theta^{n+1})$
ST	A dummy variable used to sum P^n
SU	A dummy variable used to sum V^n
T	θ^{n+1}
TLIM	Maximum value of θ^{n+1} that can be used in remaining stages
T1	Initial value of θ^{n+1} for stage 1.
U	Stage volume

Table I-1. (con't)

Symbol	Explanation
WT	R_o
X	x_1^n
XA	A dummy variable used to calculate the stage outlet BOD concentration in program SIM
XF	x_1^o
XO	Dummy variable used with x_1^n
XR	Desired X_1^N value
X1,X2,X3	Values of θ^{n+1} during Bolzano search
Y1,Y2,Y3	Values of right hand side of optimal recursion equation during Bolzano search
Z	Value of left hand side of optimal recursion equation during Bolzano search

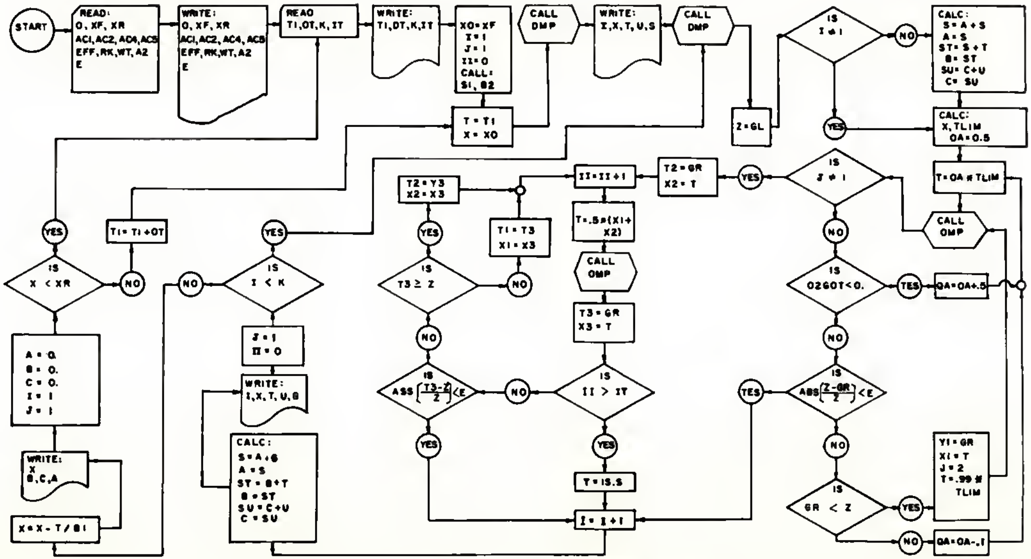

Fig. 1-2. Logic diagram for computer program OPT.

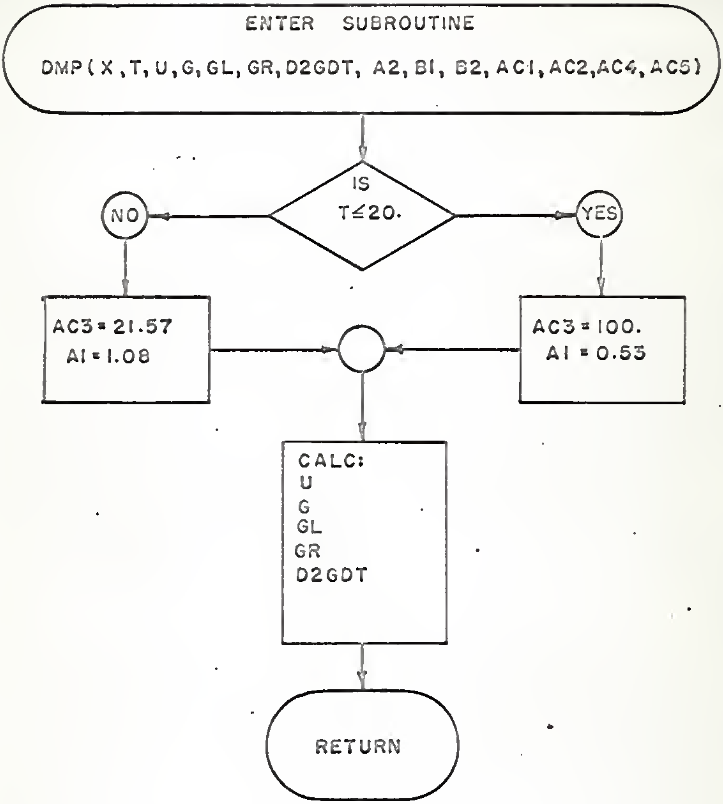

Fig. I-3. Logic diagram for subroutine DMP.

Table I-2. Computer Program OPT and Subroutine DMP

```
MON$$     JOB
MON$$     COMT 10 MINUTES, 30 PAGES,
MON$$     ASGN MJB,12
MON$$     ASGN MGO,16
MON$$     MODE GO
MON$$     EXEQ FORTRAN,,,,,,,OPT
 1 FORMAT(7E13.6)
 2 FORMAT(I3,3X,F9.6,3X,F6.1,3X,F9.0,3X,F9.0)
 3 FORMAT(2E13.6,2I3)
 4 FORMAT(6X,F9.6)
 5 FORMAT(18X,F6.1,3X,F9.0,3X,F9.0)
   READ(1,1)Q,XF,XR
   READ(1,1)AC1,AC2,AC4,AC5
   READ(1,1)EFF,RK,WT,A2
   READ(1,1)E
   WRITE(3,1)Q,XF,XR
   WRITE(3,1)AC1,AC2,AC4,AC5
   WRITE(3,1)EFF,RK,WT,A2
   WRITE(3,1)E
14 READ(1,3)T1,DT,K,IT
   WRITE(3,3)T1,DT,K,IT
   X0=XF
   I=1
   J=1
   II=0
   B1=Q/(WT*EFF)
   B2=Q/RK
22 T=T1
   X=X0
   CALLDMP(X,T,U,G,GL,GR,D2GDT,A2,B1,B2,AC1,AC2,AC4,AC5)
   WRITE(3,2)I,X,T,U,G
26 CALLDMP(X,T,U,G,GL,GR,D2GDT,A2,B1,B2,AC1,AC2,AC4,AC5)
   Z=GL
   IF(I.NE.1)GOTO35
29 S=A+G
   A=S
   ST=B+T
   B=ST
   SU=C+U
   C=SU
35 X=X-T/B1
   TLIM=B1*X
   QA=.5
38 T=QA*TLIM
39 CALLDMP(X,T,U,G,GL,GR,D2GDT,A2,B1,B2,AC1,AC2,AC4,AC5)
   IF(J.NE.1)GOTO53
   IF(D2GDT.LT.0.)GOTO46
   IF(ABS((Z-GR)/Z).LT.E)GOTO70
```

Table I-2. (Con't)

```
      IF(GR.LT.Z)GOTO48
   44 QA=QA-.1
      GOTO38
   46 QA=QA+.05
      GOTO38
   48 Y1=GR
      X1=T
      J=2
      T=.99*TLIM
      GOTO39
   53 Y2=GR
      X2=T
   55 II=II+1
      T=.5*(X1+X2)
      CALLDMP(X,T,U,G,GL,GR,D2GDT,A2,B1,B2,AC1,AC2,AC4,AC5)
      Y3=GR
      X3=T
      IF(II.GT.IT)GOTO69
      IF(ABS((Y3-Z)/Z).LT.E)GOTO70
      IF(Y3.GE.Z)GOTO66
   63 Y1=Y3
      X1=X3
      GOTO55
   66 Y2=Y3
      X2=X3
      GOTO55
   69 T=19.9
   70 I=I+1
      S=A+G
      A=S
      ST=B+T
      B=ST
      SU=C+U
      C=SU
      WRITE(3,2)I,X,T,U,G
      J=1
      II=0
      IF(I.LT.K)GOTO26
   81 X=X-T/B1
      WRITE(3,4)X
      WRITE(3,5)B,C,A
      I=1
      J=1
      A=.0
      B=.0
      C=.0
      IF(X.LT.XR)GOTO14
   90 T1=T1+DT
      GOTO22
      END
```

Table I-2. (Con't)

```
 MON$$      EXEQ FORTRAN,,,,,,,
      SUBROUTINE DMP(X,T,U,G,GL,GR,D2GDT,A2,B1,B2,AC1,AC2,AC4,AC5)
    1 FORMAT(7E13.6)
      IF(T.LE.20.)GOTO6
    3 AC3=21.57
      A1=1.08
      GOTO8
    6 AC3=100.
      A1=0.53
    8 F=B1*B2/((B1*X-T)*(B1*X-T))
      DTDT=F*X
      U=B2*T/(B1*X-T)
      G=AC1+AC2*T+AC3*T**A1+AC4*U**A2+AC5*U
      DGDT=AC2+A1*AC3*T**(A1-1.)+(AC5+A2*AC4*U**(A2-1.))*DTDT
      DTDX=B1
      DUDX=-F*T
      DGDX=DUDX*(AC5+A2*AC4*U**(A2-1.))
      GL=DGDT
      GR=DGDT+DGDX/B1
      D2UDT=F*X*2./(B1*X-T)
      G1=(A1*A1-A1)*AC3*(T**(A1-2.))
      G2=(AC5+A2*AC4*U**(A2-1.))*D2UDT
      G3=((A2*A2-A2)*AC4*(U)**(A2-2.))*DTDT
      D2GDT=G1+G2+G3
      RETURN
      END
 MON$$      EXEQ LINKLOAD
            CALL OPT
 MON$$      EXEQ OPT,MJB


      DATA
.451170E+05  .109025E-01  .109025E-02
.500000E+04  .149158E+04  .676441E+01  .009260E+00
.750000E+00  .750000E+00  .320000E+01  .666667E+00
.000050E+00
.184500E+03  .100000E+00  2 20
.139600E+03  .100000E+00  2 20
.107700E+03  .100000E+00  3 20
.081200E+03  .100000E+00  4 20
```

REFERENCES

1. Stanton, R. G., "Numerical Methods for Scientists and Engineers," Prentice-Hall, Englewood Cliffs, New Jersey (1961).

2. Rosenbrock, H. H., and C. Storey, "Computational Techniques for Chemical Engineers," Pergamon Press, London (1966).

APPENDIX II

EXPLANATION OF LAGOON SIMULATION COMPUTER PROGRAM - SIM

Computer program SIM simulates the operation of the aerated lagoon according to the aerated lagoon model used in the discrete maximum principle optimization study. The symbols used in program SIM are the same as those given in Table I-1. A logic diagram for SIM is given in Figure II-1. Program SIM is listed in Table II-1.

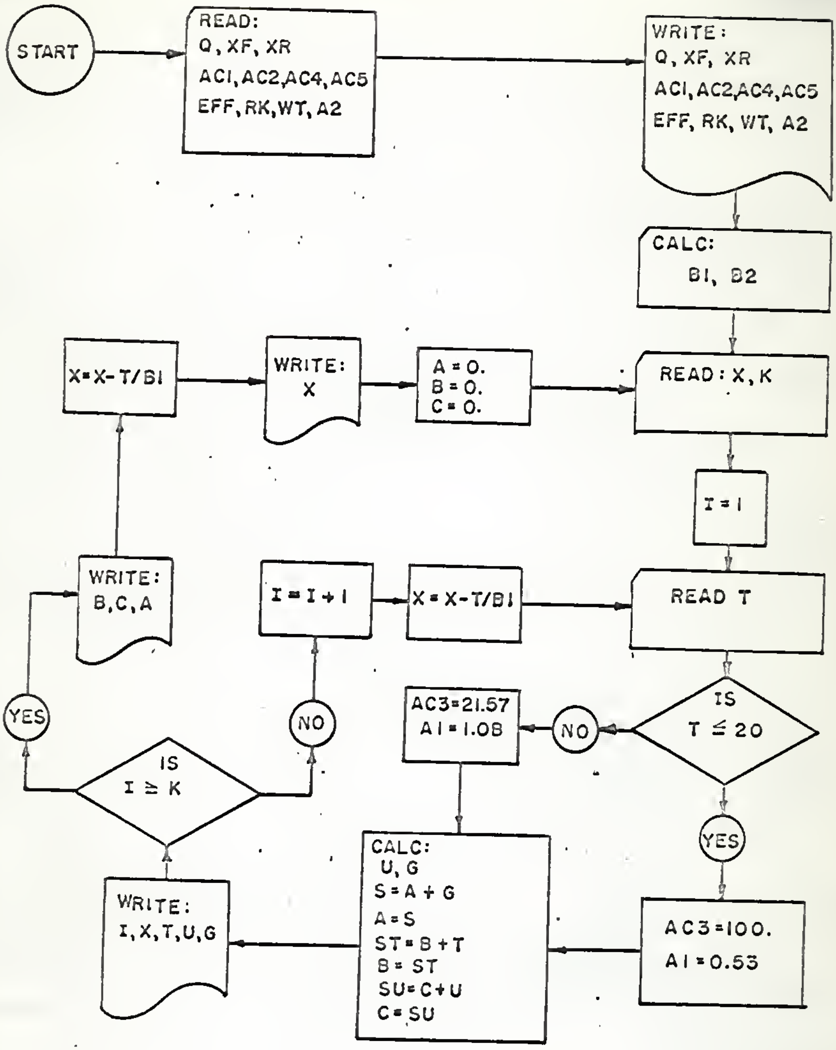

Fig. II-1. Logic diagram for computer program SIM.

Table II-1. Computer Program SIM

```
MON$$       JOB
MON$$       COMT  5 MINUTES, 10 PAGES,
MON$$       ASGN MJB,12
MON$$       ASGN MGO,16
MON$$       MODE GO
MON$$       EXEQ FORTRAN,,,,,,,SIM
  1 FORMAT(7E13.6)
  2 FORMAT(I3,3X,F9.6,3X,F6.1,3X,F9.0,3X,F9.0)
  3 FORMAT(13X,E13.6,I3)
  4 FORMAT(6X,F9.6)
  5 FORMAT(18X,F6.1,3X,F9.0,3X,F9.0)
    READ(1,1)Q,XF,XR
    READ(1,1)AC1,AC2,AC4,AC5
    READ(1,1)EFF,RK,WT,A2
    WRITE(3,1)Q,XF,XR
    WRITE(3,1)AC1,AC2,AC4,AC5
    WRITE(3,1)EFF,RK,WT,A2
    B1=Q/(WT*EFF)
    B2=Q/RK
 14 READ(1,3)X,K
    I=1
 16 READ(1,3)T
    IF(T.LE.20.)GOTO21
 18 AC3=21.57
    A1=1.08
    GOTO23
 21 AC3=100.
    A1=0.53
 23 U=B2*T/(B1*X-T)
    G=AC1+AC2*T+AC3*T**A1+AC4*U**A2+AC5*U
    S=A+G
    A=S
    ST=B+T
    B=ST
    SU=C+U
    C=SU
    WRITE(3,2)I,X,T,U,G
    IF(I.GE.K)GOTO36
    I=I+1
    X=X-T/B1
    GOTO16
 36 WRITE(3,5)B,C,A
    X=X-T/B1
    WRITE(3,4)X
    A=0.
    B=.0
    C=.0
    GOTO14
    END
```

Table II-1. (Con't)

```
   MONS$      EXEQ LINKLOAD
              CALL SIM
   MONS$      EXEQ SIM,MJB


    DATA
 .451170E+05  .109025E-01  .261660E-02
 .500000E+04  .149158E+04  .676441E+01  .009260E+00
 .750000E+00  .043700E+00  .320000E+01  .666667E+00
              .109025E-01  1
              .155800E+03  1
```

APPENDIX III

EXPLANATION OF COMPUTER PROGRAM - SEARCH

Computer program SEARCH is used to perform a one dimensional exhaustive search on the two stage aerated lagoon model given by Equations (28) and (29). Table III-1 explains the symbols used in SEARCH which have not previously been explained in Table I-1. The logic diagram for SEARCH is given in Figure III-1. Program SEARCH is listed in Table III-2.

Table III - 1. Explanation of the Symbols Used In Computer Program SEARCH

Symbol	Explanation
A	Constant used in the calculation of the incremental change in the search variable X(2)
AK	k^M
B	Constant used in the calculation of the incremental change in the search variable X(2)
CK	K
C(1),C(2),C(3)	Concentration of micro-organisms going into the first stage, in the first ideal back-mix stage, and in the second ideal back-mix stage, respectively.
D	Constant used in the calculation of the incremental change in the search variable X(2)
DEL	Incremental change in the second variable X(2)
DK	k_D
G(2), G(3)	Cost of stage one and two, respectively
P(2), P(3)	Aerator motor size of stage one and two, respectively
SG	Total cost of process

Table III-1. (Continued)

Symbol	Explanation
SP	Total horsepower of both aerator motors used in the process
ST	Sum of holding times for both stages
SV	Sum of volumes for both stages
T(2), T(3)	Holding time for stage one and two, respectively
V(2), V(3)	Volume of stage one and two, respectively
X(1), X(2), X(3)	Concentration of BOD going into stage one, in stage one, and in stage two, respectively
Y	Y

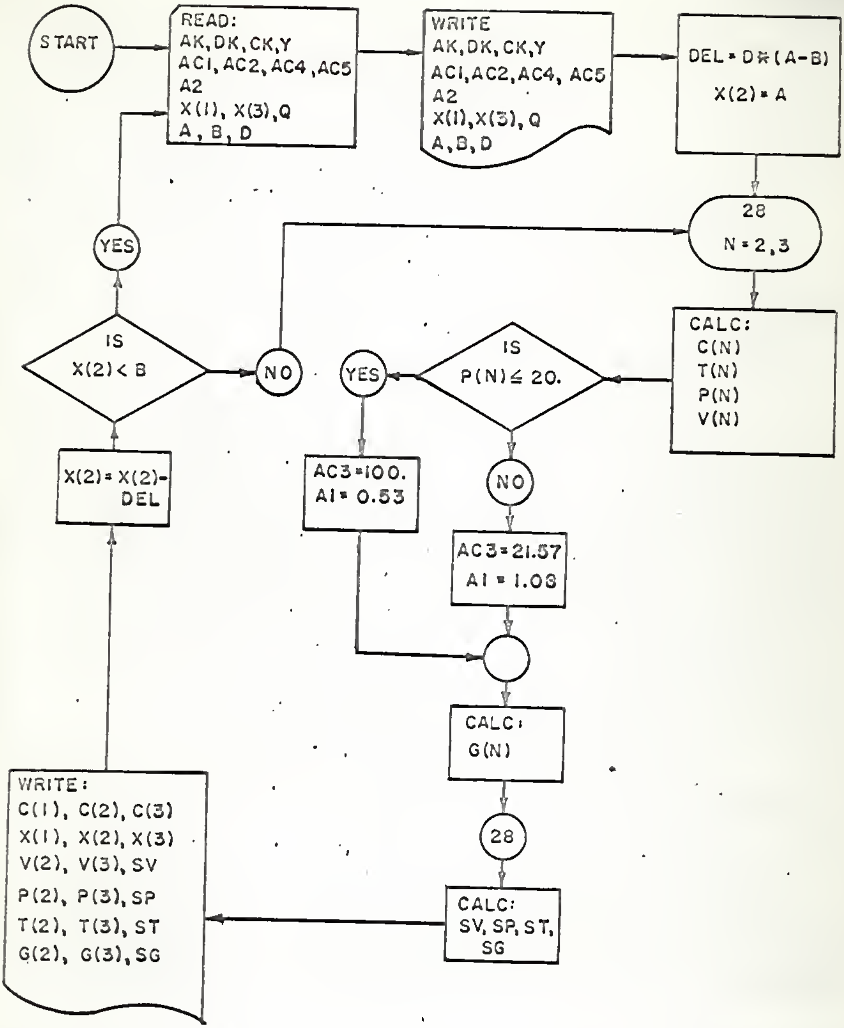

Fig. III-1. Logic diagram for computer program SEARCH.

Table III-2. Computer Program SEARCH

```
MONSS     JOB
MONSS     COMT  5 MINUTES,  5 PAGES,
MONSS     ASGN MJB,12
MONSS     ASGN MGO,16
MONSS     MODE GO
MONSS     EXEQ FORTRAN,,,,,,,SEARCH
      DIMENSIONX(3),C(3),T(3),G(3),P(3),V(3)
    1 FORMAT(7E13.6)
    2 FORMAT(1HL,13X,3E13.6)
    3 FORMAT(1HK,26X,3E13.6)
    4 FORMAT(1HK,13X,3E13.6)
    5 FORMAT(1H1,7F13.6)
    6 READ(1,1)AK,DK,CK,Y
      READ(1,1)AC1,AC2,AC4,AC5,A2
      READ(1,1)X(1),X(3),Q
      READ(1,1)A,B,D
      WRITE(3,5)AK,DK,CK,Y
      WRITE(3,1)AC1,AC2,AC4,AC5,A2
      WRITE(3,1)X(1),X(3),Q
      WRITE(3,1)A,B,D
      DEL=D*(A-B)
      X(2)=A
   16 DO28N=2,3
      C(N)=C(N-1)+((X(N-1)-X(N))*(AK*X(N)-DK*(CK+X(N)))*Y)/(AK*X(N))
      T(N)=((X(N-1)-X(N))*Y*(CK+X(N)))/(AK*X(N)*C(N))
      P(N)=Q*(X(N-1)-X(N))*2.59583E-05
      V(N)=Q*T(N)
      WRITE(3,3) C(N),T(N),P(N)
      IF(P(N).LE.20.)GOTO25
      AC3=21.57
      A1=1.08
      GOTO27
   25 AC3=100.
      A1=0.53
   27 G(N)=AC1+AC2*P(N)+AC3*P(N)**A1+AC4*V(N)**A2+AC5*V(N)
   28 CONTINUE
      SV=V(2)+V(3)
      SP=P(2)+P(3)
      ST=T(2)+T(3)
      SG=G(2)+G(3)
      WRITE(3,2)C(1),C(2),C(3)
      WRITE(3,4)X(1),X(2),X(3)
      WRITE(3,3)V(2),V(3),SV
      WRITE(3,3)P(2),P(3),SP
      WRITE(3,3)T(2),T(3),ST
      WRITE(3,3)G(2),G(3),SG
      X(2)=X(2)-DEL
      IF(X(2).LT.B)GOTO6
      GOTO16
      END
```

Table III-2. (Con't)

```
   MONSS       EXEQ LINKLOAD
               CALL SEARCH
   MONSS       EXEQ SEARCH,MJB
     .

     DATA
.100000E+00   .002000E+00   .100000E+03   .500000E+00
.500000E+04   .149158E+04   .676441E+01   .009260E+00   .666667E+00
.175000E+03   .001750E+03   .451170E+05
.024000E+03   .010000E+03   .100000E+00
.150000E+03   .010000E+03   .050000E+00
.174990E+03   .001755E+03   .050000E+00
```

OPTIMIZATION STUDIES OF TWO

WATER PURIFICATION SYSTEMS

by

THOMAS DEAN DENCHFIELD

B. S., KANSAS STATE UNIVERSITY, 1965

AN ABSTRACT OF A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Chemical Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1967

This work illustrates the application of optimization techniques to water purification processes. Two different types of water purification processes are studied. A different optimization technique is used to optimize each of the processes.

In Part I, the Process Optimization Program (POP-II), a nonlinear programming technique, is used to optimize a Multi-Effect Multi-Stage (MEMS) seawater distillation process.

The results of the optimization are in good agreement with the results obtained by applying a discrete version of Pontryagin's Maximum Principle. At optimum operating conditions, the one thousand gallon per hour MEMS process optimized, produces potable water at a cost of \$0.2866 per one thousand gallons.

Complex optimization problems with many independent variables can be rapidly programmed for solution by POP-II because the first partial derivatives which POP-II requires are calculated by the central differences numerical approximation technique rather than by analytic methods. However, some manipulation of the input data and several trials to select the proper operating mode may be required before an optimum is obtained by POP.

In Part II, a simplified model of a stage wise aerated lagoon process for the partial purification of petroleum refinery waste water is derived. This model was optimized using a discrete version of Pontryagin's Maximum Principle. A method of working with piecewise smooth objective functions was used to obtain the optimum solution.

The optimization study predicts, in agreement with theory, that tapered aeration is the best mode of operation for stagewise operated aerated lagoons. However, to assure better quantitative results, a more comprehensive aerated lagoon model is proposed for future work.